Universidade de Évora

Escola de Ciências e Tecnologia

Mestrado em Engenharia Informática

Extracção de relações entre entidades mencionadas

João Manuel dos Santos Sequeira

Orientador
Teresa Cristina de Freitas Gonçalves

Co-Orientador
Paulo Miguel Torres Duarte Quaresma

Évora, Dezembro de 2011

Mestrado em Engenharia Informática

Extracção de relações entre entidades mencionadas

João Manuel dos Santos Sequeira

Orientador
Teresa Cristina de Freitas Gonçalves

Co-Orientador
Paulo Miguel Torres Duarte Quaresma

Sumário

Actualmente existe uma grande quantidade de conteúdos digitais de cariz académico, pessoal e noticioso, entre outros, disponíveis para consulta na Internet. A obtenção de informação estruturada a partir destes conteúdos de forma manual tornou-se praticamente impossível. Assim, nos últimos anos tem-se registado a um aumento na investigação de sistemas para análise e extracção de informação de forma automática.

A classificação dos documentos por temas ou categorias constitui uma forma de relacionar conteúdos. No entanto, os documentos poderão, de igual forma, ser relacionados a partir das entidades que neles figuram, sejam elas Pessoas, Locais ou Organizações; mais ainda, ao extrair informação sobre as relações existentes entre as entidades, as formas de interacção entre documentos tornam-se muito mais ricas já que será possível, por exemplo, relacionar os documentos que referem que determinada entidade praticou determinada acção e quais as entidades que a sofreram.

Este trabalho propõe um sistema para identificação e extracção de relações entre entidades presentes num documento. As relações são obtidas a partir de um classificador de argumentos sintácticos utilizado em conjunto com um reconhecedor de entidades.

Tratando-se de um sistema aplicado à língua Portuguesa foi necessário o desenvolvimento de alguns recursos específicos para a língua: um etiquetador de categorias gramaticais e dois corpora: um para ser utilizado pelo etiquetador e outro com informação sintáctica a nível das palavras, sintagmas e orações para ser utilizado na tarefa de classificação de argumentos sintácticos.

Embora utilizando um classificador de argumentos sintácticos preliminar, a experimentação mostra que o sistema desenvolvido consegue atingir o objectivo proposto e identificar relações entre entidades. Por outro lado, a criação dos recursos referidos vem enriquecer o conjunto de ferramentas disponíveis para a língua Portuguesa passíveis de serem utilizados em futuros trabalhos.

Palavras-chave: processamento de linguagem natural, classificação de argumentos sintácticos, reconhecimento de entidades, etiquetador de categorias gramaticais, corpora para a língua Portuguesa.

Extraction of relations between named entities

Abstract

Currently there is a large amount of digital content, being personal, academic and news, among others, available on the Internet. Obtaining structured information from these contents by hand has become virtually impossible. So, in recent years there has been an increase in the investigation of systems for automatic analysis and information extraction.

Classification of documents by themes or categories is a way of relating content. However, documents can, likewise, be related by the entities they contain, being they people, places or organizations; moreover, extracting information on relations between the entities, the forms of interaction between documents become much richer as it will enable, for example, to list the documents that refer to a particular entity having practiced a specific action and which entities have suffered that action.

This paper proposes a system for identifying and extracting relations between entities present in a document. Relations are obtained from a semantic role labeller used in conjunction with named entity recognizer.

Being applied to the Portuguese language, it was necessary to develop specific resources for the language: a part-of-speech tagger and two corpora: one to be used with the POS-tagger and other with syntactic information for words, phrases and sentences to be used by the semantic role labeller.

Although a preliminary semantic role labeller, experimentation shows that the system can achieve the proposed objective and identify relationships between entities. On the other hand, the creation of the refered resources will enrich the available Portuguese language set of tools that can be used in future work.

Keywords: Natural language processing, semantic role labelling, named entity recognizer, part-of-speech tagger, corpora for the Portuguese language

À minha família e a todos os que me apoiam.

Agradecimentos

Desde já agradeço à professora Teresa Gonçalves e ao professor Paulo Quaresma por me terem orientado e apoiado ao longo de todas as fases deste trabalho. Destaco especialmente o tempo despendido, a paciência, confiança e as valiosas contribuições que permitiram que este trabalho se tornasse uma realidade.

Quero agradecer também ao meu Pai, Eduardo Sequeira, Mãe, Rosa Sequeira, Irmã, Elsa Sequeira e restantes familiares que me apoiaram em todos os momentos desta minha jornada universitária.

Um agradecimento muito especial à Mara Pereira, onde eu sempre encontrei palavras de motivação e apreço que nunca me deixaram desistir dos meus objectivos.

Agradeço à Janete Nunes, Nádia Rodrigues e Maria Isabel pelas vivências inesquecíveis no 28.

Agradeço ao Nuno Miranda pelo tempo despendido no auxílio na elaboração deste trabalho e pelo apoio durante o projecto que precedeu o mesmo. À ViaTecla que me deu a oportunidade de realizar um projecto em ambiente empresarial.

Não vou agradecer aos meus colegas, mas sim aos amigos que fiz e me acompanharam ao longo deste curso tornando os momentos de trabalho e de convívio em momentos únicos.

Acrónimos

XML Extensible Markup Language

UE Universidade de Évora

CONLL Conference on Computational Natural Language Learning

SIGNLL Special Interest Group on Natural Language Learning

SVM Support Vector Machines

ILP Integer Linear Programming

CRF Conditional Random Fields

CMM Conditional Markov Models

KNN k-Nearest Neighbor

IBL Instance-Based Learning

Span Conjunto de palavras

POS Part-of-Speech

IOB Inside-Outside-Begin

UPC Universidade Politécnica da Catalunha

TALP Center for Language and Speech Technologies and Applications

CFG Context-Free Grammar

CCG Combinatory Categorial Grammar

LFG Lexical Functional Grammar

Conteúdo

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Actualmente existe uma grande quantidade de conteúdos digitais de cariz académico, pessoal e noticioso entre outros disponíveis para consulta na Internet. A tarefa de obter informação de conteúdos não tratados de fontes tão dispares manualmente tornou-se praticamente impossível [62].

Em 2010 existiam 2000 milhões de utilizadores na Internet [2], ou seja, sensivelmente 28% duma população mundial de quase 7000 milhões [107].

O aumento da utilização da Internet deveu-se em muito à expansão da componente social e da partilha de conteúdos, permitindo assim que os conteúdos ficassem disponíveis em tempo real para posteriores críticas por parte da comunidade e alterações (se necessário) por parte dos autores ou outros intervenientes.

Com o incremento de conteúdos textuais em formato digital existiu também um aumento na investigação de sistemas que os consigam analisar e que consigam extrair informação automaticamente. Assim, nos últimos anos tem existido uma crescente procura de aplicações de processamento de linguagem natural[1] (PLN) [13].

A classificação de argumentos sintácticos[2] constitui uma área de grande interesse, devido à sua crescente importância em sistemas de extracção de informação, pergunta-resposta, sumarização de documentos entre outras aplicações que necessitam de informação semântica [12].

[1] Do inglês, *Natural Language Processing* (NLP).

[2] O termo inglês *Semantic Role Labelling* (SRL) é utilizado nas conferências internacionais, retratando a identificação das relações semânticas entre os diferentes constituintes duma frase. Utiliza-se a denominação presente na Linguateca; outras possíveis denominações: argumentos semânticos ou papeis semânticos.

Esta tarefa do processamento de linguagem natural possui vários recursos disponíveis para a língua Inglesa, produto de vários projectos apresentados ou realizados no âmbito de conferências internacionais [12]. Existe, no entanto, muita matéria a ser explorada no âmbito de outras línguas, estando a Portuguesa entre elas.

Outra área do processamento de língua natural com uma diversidade de trabalhos desenvolvido é a do reconhecimento de entidades mencionadas[3] Esta tarefa consiste em extrair nomes de entidades, tais como Pessoas, Locais ou Organizações dos textos [84].

Para a língua Portuguesa existem diversas aproximações a esta tarefa. Algumas estão disponíveis na página da Linguateca [53] tais como o Rembrandt[4] desenvolvido na Faculdade de Ciências da Universidade de Lisboa e o SMELL[5] desenvolvido no Laboratório de Engenharia da Linguagem no Instituto Superior Técnico.

1.1 Motivação

O tema deste trabalho foi motivado pelo objectivo do projecto em que ele se insere: a identificação de relações entre entidades para permitir a navegação inteligente entre documentos onde essas entidades estão presentes.

O projecto foi desenvolvido no âmbito do QREN TV.COMmunity [78], para a região do Alentejo, no laboratório de Excelência .NET na Universidade de Évora. Este laboratório surgiu de um protocolo tripartido entre a empresa ViaTecla [102], a Universidade de Évora [100] e a Microsoft [61]. Deste projecto faz parte o sistema reconhecedor de entidades mencionadas documentado em [62, 63].

As relações entre entidades são identificadas a partir da extracção de informação referente a entidades mencionadas, tais como Pessoas, Locais e Organizações e a argumentos sintácticos das frases, tais como sujeito e complemento directo.

Uma segunda motivação para a realização deste trabalho é a investigação numa área em crescente expansão – o processamento de língua natural – utilizando técnicas de aprendizagem e a sua aplicação à língua Portuguesa, disponibilizando, assim, novos recursos (sejam eles corpora ou aplicações) para futuros estudos.

1.2 Objectivos

Como já foi referido, devido à existência de uma grande quantidade de textos em formato digital a extracção de informação de forma manual é incomportável. Assim,

[3] Do termo inglês *Named Entities Recognition* (NER). Um sinónimo aos termos, entidades mencionadas ou entidades, usados neste trabalho pode também ser entidades nomeadas.

[4] Disponível em `http://xldb.di.fc.ul.pt/Rembrandt/?do=home`.

[5] Disponível em `http://label.ist.utl.pt/pt/smell_intr_pt.php`.

aplicações como as apresentadas neste trabalho são importantes para tratar e manter a informação dos textos evitando a necessidade de abordagens manuais, com o trabalho e as perdas de tempo a elas associadas.

A identificação de relações entre entidades presentes em textos permite novas formas inteligentes de navegação entre documentos tais como:

- relacionar documentos em que uma determinada entidade aparece;
- relacionar entidades ligadas por determinada acção em diferentes documentos;
- navegar entre documentos em que uma determinada entidade pratica acções e verificar as entidades que as sofreram;
- navegar entre documentos em que uma determinada entidade sofre acções e analizar as entidades que as praticaram.

Este trabalho pretende atingir os seguintes objectivos:

- por um lado, analisar os corpora e ferramentas existentes para o processamento da língua Portuguesa;
- por outro, examinar algoritmos e métodos de aprendizagem automática passíveis de serem utilizados na criação de modelos para a classificação de argumentos sintácticos;
- com base nas análises efectuadas, desenvolver as aplicações auxiliares necessárias para a tarefa de classificação de argumentos sintácticos, nomeadamente um etiquetador de categorias gramaticais;
- e ainda construir corpora para a língua Portuguesa com as características linguísticas usadas internacionalmente para a tarefa de classificação de argumentos sintácticos;
- finalmente, utilizando o corpus e as aplicações auxiliares, implementar uma versão preliminar de um classificador de argumentos sintácticos para a língua Portuguesa e comparar os seus resultados com outros obtidos internacionalmente para a mesma tarefa aplicada à língua Inglesa.

1.3 Abordagem proposta

Para atingir os objectivos propostos usaram-se corpora para a língua Portuguesa e para a Inglesa. Para a língua Portuguesa foi utilizada a componente `CETÉMPublico` do corpus Bosque[6] [3]; o `CETÉMPublico` é composto por notícias em Português

[6]Disponível em: `http://www.linguateca.pt/Floresta/corpus.html`.

Europeu do jornal Publico [75]. Para a língua Inglesa foi usado o corpus `Penn Treebank`[7] [58, 59] da tarefa conjunta da edição de 2004 da conferência CoNLL (*Computational Natural Language Learning*) [12].

Com base no corpus `CETÉMPublico` foi implementado um etiquetador de categorias gramaticais. Este etiquetador foi posteriormente utilizado como fonte de informação para o sistema reconhecedor de entidades mencionadas e para a construção do corpus da tarefa de classificação de argumentos sintácticos. Este corpus foi construído tendo como referência a representação utilizada em tarefas conjuntas internacionais.

Utilizando um classificador sequencial abordou-se o problema da classificação de argumentos sintácticos, tanto para a língua Portuguesa como para a língua Inglesa. O classificador para a língua Inglesa foi construído com o objectivo de comparar os seus resultados com os valores obtidos em conferências internacionais.

Com a identificação das entidades e a classificação dos argumentos sintácticos implementou-se uma ferramenta capaz de extrair relações entre as entidades.

1.4 Principais contribuições

Este trabalho elabora um estudo da tarefa de classificação de argumentos sintácticos e constrói um classificador preliminar aplicado à língua Portuguesa. Este classificador é depois utilizado para extrair relações entre entidades mencionadas.

Ao longo do seu desenvolvimento foram criados diversos recursos e modelos para o processamento de linguagem natural para a língua Portuguesa:

- um corpus para o desenvolvimento de etiquetadores de categorias gramaticais;
- um etiquetador de categorias gramaticais [63];
- um corpus com informação gramatical, sintáctica e semântica para o desenvolvimento de classificadores de argumentos sintácticos, tendo como referência os corpora utilizados nas conferências internacionais CoNLL;
- um classificador de argumentos sintácticos preliminar [90].

Com estes recursos e modelos foi possível desenvolver uma aplicação protótipo capaz de analisar textos escritos na língua Portuguesa e extrair as relações existentes entre entidades mencionadas.

Enquanto o etiquetador produziu resultados equivalentes aos obtidos com outros etiquetadores e outras línguas, o classificador de argumentos sintácticos preliminar, que apenas utiliza informação sintáctica ao nível das orações, mostrou ter um menor desempenho que o demonstrado nas conferências internacionais CoNLL.

[7] Página do projecto: `http://www.cis.upenn.edu/~treebank/`.

1.5 Organização da dissertação

Esta dissertação está organizada da seguinte maneira:

- no Capítulo 2 são introduzidas as noções básicas da aprendizagem automática e os algoritmos e métodos mais utilizados na construção de etiquetadores de categorias sintácticas e classificadores de argumentos sintácticos. Incluem-se os algoritmos máquinas de vectores de suporte, campos condicionais aleatórios e modelos condicionais de Markov e os métodos de marcação amplamente utilizados no processamento de linguagem natural como o método IOB. Por fim, são introduzidas as medidas de avaliação de desempenho dos sistemas (precisão, cobertura e F_1);

- no Capítulo 3 é abordada a aproximação proposta para extracção de relações entre entidades. É feita a caracterização do Bosque 8.0, o corpus original do qual foi extraída a informação necessária para a criação dos corpora especializados e são introduzidas as ferramentas para processamento de língua natural desenvolvidas: o etiquetador de categorias sintácticas e o classificador de argumentos sintácticos;

- o Capítulo 4 foca o etiquetador de categorias gramaticais multi-linguístico. São introduzidas ferramentas específicas para esta tarefa, é descrita a construção do corpus utilizado para criação do modelo para a língua Portuguesa e é apresentado o desempenho obtido pelo etiquetador;

- o Capítulo 5 introduz o classificador de argumentos sintácticos. São introduzidas as ferramentas utilizadas, são descritos corpora construídos e é apresentado o desempenho do classificador preliminar para os corpora da língua Portuguesa e da língua Inglesa, realizando a comparação do desempenho com os dos sistemas participantes na tarefa conjunta do CoNLL'04.

- no Capítulo 6 são discutidos os resultados obtidos e apresentadas as conclusões e o trabalho futuro com o objectivo de melhorar os classificadores e corpora utilizados neste trabalho.

Capítulo 2

Aprendizagem de Língua Natural

A aprendizagem automática[1] é uma área da inteligência artificial que estuda o desenvolvimento de sistemas com habilidade de produzir conhecimento a partir de dados. Esse conhecimento é depois utilizado para realizar tarefas sobre novos dados [56], ou seja, são explorados métodos de programação que permitem aos computadores aprender a efectuar determinadas tarefas [25].

A aprendizagem de língua natural[2] tem como objectivo a criação de modelos para as mais diversas tarefas de processamento de língua natural utilizando técnicas de aprendizagem automática.

Este capítulo apresenta a área da aprendizagem automática a a sua aplicação ao processamento de língua natural. A secção 2.1 introduz os conceitos básicos da aprendizagem automática, a secção 2.2 descreve os algoritmos utilizados neste trabalho e a secção 2.3 define as medidas de desempenho utilizadas na avaliação dos sistemas. A secção 2.4 introduz a aplicação da aprendizagem automática à área da língua natural.

2.1 Aprendizagem supervisionada e não supervisionada

A aprendizagem automática pode dividir-se entre aprendizagem não supervisionada e supervisionada [25]. Ao contrário da aprendizagem supervisionada, na apren-

[1] Do Inglês, *machine learning*.
[2] Do inglês, *natural language learning*.

dizagem não supervisionada são fornecidos dados não classificados, ou seja, não é fornecida informação de classes a aprender. O seu objectivo é encontrar padrões nos dados de modo a criar hipóteses para a classificação de futuros dados. Uma das técnicas mais comuns deste tipo de aprendizagem é o agrupamento[3] [34].

Na Figura 2.1 é apresentado um exemplo do resultado final da aplicação da técnica de agrupamento a um espaço com três tipos de instâncias: triângulos, quadrados e círculos; o objectivo consiste em agrupar as instâncias de cada classe.

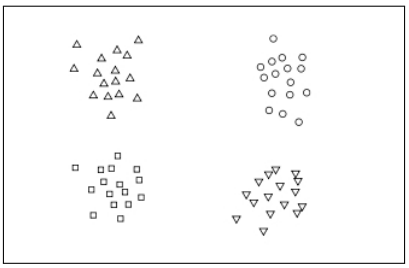

Figura 2.1: Agrupamento: o objectivo é agrupar os objectos segundo a sua classe (triângulo, quadrado e círculo). (Fonte: [97])

Na aprendizagem supervisionada são fornecidos conjuntos de dados classificados para realizar o treino da função desejada, criando um modelo para usar como classificador [56]. O treino do classificador é realizado com dados na forma `(exemplo,classe)`, ou seja, um elemento do conjunto é um par (x_i, y_i); o modelo gerado pode ser visto como uma função f que recebe um qualquer novo exemplo x e dá como resultado uma classe y [57].

Na Figura 2.2 está presente a arquitectura de criação de um classificador: este é treinado a partir de um conjunto de dados em que cada instância é composta pelos seus atributos e respectiva classe.

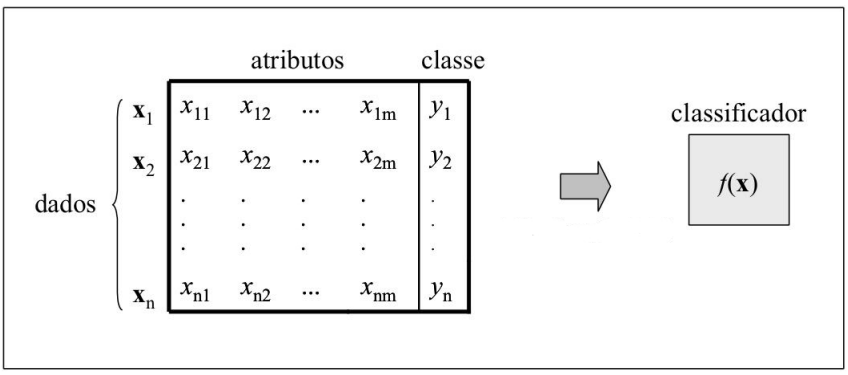

Figura 2.2: Criação do classificador a partir de um conjunto com n elementos, de x_1 a x_n; cada elemento tem m atributos e uma classe associada. (Fonte: [57])

[3] Do Inglês, *clustering*.

2.2 Algoritmos de aprendizagem supervisionada

Vários algoritmos de aprendizagem automática têm sido utilizados nas diferentes tarefas de processamento de língua natural. As avaliações conjuntas desenvolvidas no âmbito das conferências CoNLL apresentam bons exemplos dessa variedade de aproximações:

- na tarefa de classificação de argumentos semânticos (edições dos anos 2004 [12] e 2005 [13]), os algoritmos que obtiveram melhores desempenhos foram: máquinas de vectores de suporte[4], Winnow e programação linear de inteiros[5].
- na tarefa de reconhecimento de entidades mencionadas (edições de 2002 [84] e 2003 [23]) os melhores desempenhos foram obtidos com os algoritmos: AdaBoost aplicado a árvores de decisão de profundidade fixa e modelos de máxima entropia[6].

Os algoritmos implementados nas ferramentas utilizadas neste trabalho foram: máquinas de vectores de suporte, campos condicionais aleatórios[7] e modelos condicionais de Markov[8].

Nas sub-secções seguintes introduzem-se estes algoritmos.

2.2.1 Máquinas de vectores de suporte (SVM)

Os classificadores lineares separam os dados por meio de hiperplanos. Quando nos dados de treino estão presentes duas classes, positiva e negativa, sendo x_i um vector de características e y_i a classe do exemplo i [49], tem-se

$$\begin{gathered}(x_i, y_i), \ldots, (x_n, y_n) \\ x_i \in \mathbb{R}^n \\ y_i \in \{+1, -1\}\end{gathered} \tag{2.1}$$

Assim, a função de um hiperplano para separar os exemplos positivos dos negativos [49] é dada por

$$(w \cdot x) + b = 0 \qquad w \in \mathbb{R}^n, b \in \mathbb{R} \tag{2.2}$$

[4] Do Inglês, *support vector machines* (SVM).
[5] Do Inglês, *integer linear programming* (ILP).
[6] Do inglês, *Maximum Entropy Models*.
[7] Do Inglês, *condicional random fields* (CRF).
[8] Do Inglês, *conditional Markov models* (CMM).

As máquinas de vectores de suporte são classificadores lineares que procuram obter o hiperplano 'óptimo', ou seja, os valores de w e b que permitem separar os dados maximizando a sua margem sem erros de classificação. A Figura 2.3 mostra um exemplo de dois possíveis hiperplanos que separam um conjunto de dados e as respectivas margens.

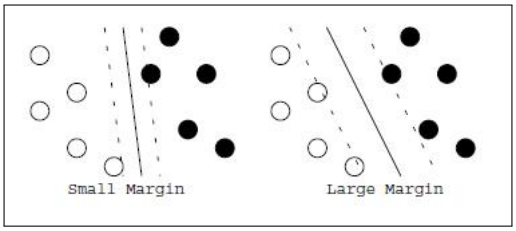

Figura 2.3: Hiperplanos de separação (linha continua) e as respectivas margens (linhas tracejadas); margem menor à esquerda, margem maior à direita. (Fonte: [49])

O processo de aprendizagem das máquinas de vectores de suporte consiste em construir classificadores lineares num novo espaço de características (transformação do espaço de entrada) induzido pela função de núcleo escolhida e que corresponde a fronteiras de decisão não lineares no espaço de entrada.

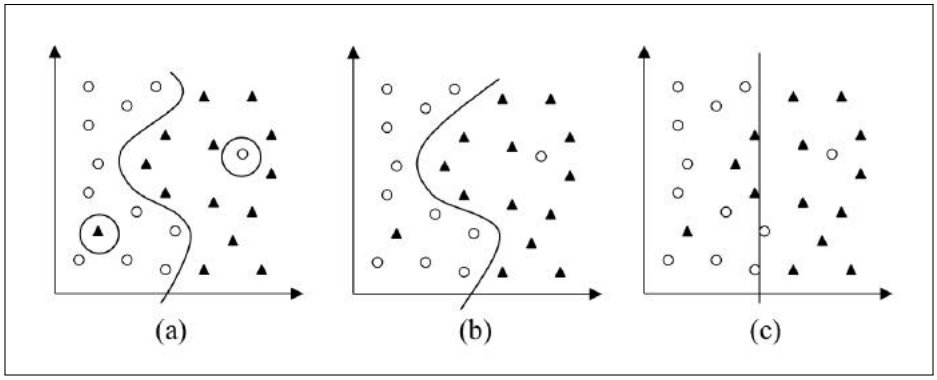

Figura 2.4: Fronteiras de decisão obtidas por diferentes classificadores. (Fonte: [57])

As máquinas de vectores de suporte permitem fazer um compromisso entre classificadores simples que produzem diversos erros e classificadores sem erros mas de grande complexidade:

- em (a) todos os exemplos são classificados correctamente, até mesmo dois exemplos que possivelmente são ruído. Sendo um classificador tão específico é provável que cometa um maior número de erros quando receber dados novos;

- o exemplo (b) representa um classificador de complexidade mediana classificando a maior parte dos dados correctamente mas sem se focar em demasia no ruído;

- em (c) não são considerados exemplos de classes opostas que estejam muito próximos. Este classificador também comete muitos erros devido à sua pouca ajustabilidade às classes.

As características que tornam as máquinas de vectores de suporte tão atractivas nas tarefas de classificação são: uma base teórica bem definida [101], robustez quando são usados objectos de grandes dimensões e boa capacidade de generalização dos classificadores quando expostos a novos dados que não os utilizados para o seu treino [101, 56].

2.2.2 Modelos condicionais de Markov (CMM)

Os modelos condicionais de Markov, também conhecidos como modelos de máxima entropia de Markov[9] [60, 72], são baseados em modelos escondidos de Markov[10] [80].

Um modelo escondido de Markov é um autómato de estados finitos com observações e transições estocásticas [79]. O funcionamento baseia-se na sequência de transições produzida a partir de um estado inicial; ao ser emitida uma observação num determinado estado é realizada a transição para o estado seguinte onde é emitida nova observação até ser alcançado um estado designado como final [60].

A definição formal é dada em [60]. Um modelo escondido de Markov é dado por um conjunto finito de estados S, um conjunto de observações O, uma distribuição do estado inicial $P_0(s)$ e duas distribuições de probabilidade condicionada

- para a transição do estado s' para o estado s, $P(s|s')$, sendo s e $s' \in S$;
- para a observação o, $P(o|s)$, onde $o \in O$ e $s \in S$.

Um exemplo da utilização de modelos escondidos de Markov é a extracção de informação. Neste problema existe uma sequência de etiquetas $l_1...l_m$ atribuídas a cada sequência de observações $O_1...O_m$; a cada nova observação o objectivo final é obter a sequência de etiquetas mais provável [60].

Ao contrário dos modelos escondidos onde são usadas duas distribuições de probabilidade – de transição e observação – nos modelos condicionais é usada apenas a distribuição de probabilidade $P(s|s', o)$ para o estado actual s condicionada ao estado anterior s' e à observação actual o, ou seja, obtém-se a probabilidade da transição do estado s' para s devido à observação o [60].

A Figura 2.5 mostra graficamente a diferença entre o modelo escondido e o modelo condicional de Markov.

[9] Do Inglês, *maximum-entropy Markov model (MEMM)*.
[10] Do Inglês, *hidden Markov models (HMM)*.

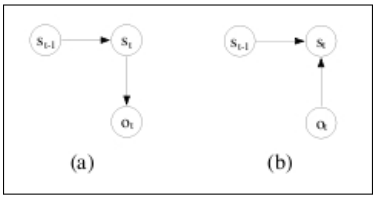

Figura 2.5: Estrutura gráfica de dependências: (a) modelo escondido de Markov (b) modelo condicional de Markov. (Fonte: [60])

A máxima entropia baseia-se no princípio de que o melhor modelo obtido deve ser consistente com certas restrições derivadas dos dados de treino e com o menor número de suposições sobre o mesmo [104]. Para a resolução destas restrições é utilizada a distribuição de máxima verosimilhança

$$P_{s'}(s|o) = \frac{1}{Z(o,s'))} exp(\sum_{a} \lambda_a f_a(o,s)) \tag{2.3}$$

onde λ_a são parâmetros a aprender pelo método de escalonamento interactivo generalizado[11] [22], $Z(o,s')$ é o factor de normalização e $f_a(o,s)$ é função da observação o e de um novo estado s.

Em $f_a(o,s)$, a é um par ordenado $a=(b,s)$, onde b é uma característica binária da observação e s o estado de destino dessa observação [104] num determinado momento no tempo t, dada pela função

$$F_{(b,s)}(o_t,s_t) = \begin{cases} 1 & \text{se } b_{o_t} \text{ é verdade e s= } s_t \\ 0 & \text{caso contrário} \end{cases} \tag{2.4}$$

2.2.3 Campos condicionais aleatórios (CRF)

Um campo condicional aleatório é um modelo gráfico não direccionado usado para definir a distribuição de probabilidade conjunta de sequências de etiquetas dado um conjunto de sequências de observações [105]. Os campos condicionais aleatórios possuem como base os modelos condicionais de Markov e as suas definições [51].

A definição formal é dada em [51]. Seja $G=(V,E)$ um grafo tal que $Y=(Y_v)_{v\in V}$, com Y indexado pelos vértices de G; então (X,Y) é um possível campo condicional aleatório quando condicionado em X, e as variáveis aleatórias Y_v obedecem à propriedade de Markov

$$p(Y_v|X,Y_w,w\neq v) = p(Y_v|X,Y_w,w\sim v) \tag{2.5}$$

onde $w\sim v$ indica que w e v são vizinhos em G.

[11] Do Inglês, *generalised iterative scaling* (GIS).

A Figura 2.6 apresenta a estrutura gráfica de dependências de um campo condicional aleatório.

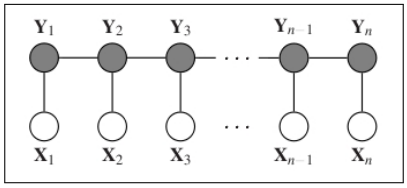

Figura 2.6: Estrutura gráfica dum campo condicional aleatório para sequências. (Fonte: [104])

Os campos condicionais aleatórios podem ser aplicados às mais variadas tarefas de processamento de linguagem natural. Por exemplo numa tarefa de classificação gramatical onde X representa as frases, Y as etiquetas gramaticais das frases e γ o conjunto de possíveis etiquetas e, dada uma sequência de observações X, para cada etiqueta particular de uma sequência Y é usada a expressão [51]

$$exp(\sum_{e \in E,K} \lambda_k f_k(e, y|_e, x) + \sum_{v \in V,k} \mu_k g_k(v, y|_v, x)) \tag{2.6}$$

onde x é uma sequência de dados, y uma sequência de classificação, $y|_s$ é o conjunto de componentes de y associado aos vértices no sub-grafo S, λ_k e μ_k são parâmetros estimados a partir dos dados de treino.

2.3 Medidas de desempenho

A avaliação dos sistemas de classificação automática é realizada através de medidas do seu desempenho e eficiência [25]. A eficiência mede o tempo necessário quer na construção do modelo, quer na classificação de novos exemplo.

O desempenho é medido através do cálculo da precisão[12] (π), cobertura[13] (ρ) e da medida F_β[14] para cada classe. Estes cálculos utilizam uma matriz de confusão que considera positiva (+) a classe em estudo e negativa (−) as restantes classes e onde é realizada a comparação entre as classes a que os exemplos pertencem (classe correcta) e aquelas que o algoritmo diz pertencer (classe prevista).

A Tabela 2.1 mostra uma matriz de confusão: VP representa o nº de verdadeiros positivos, FP o nº de falsos positivos, FN o nº de falsos negativos e VN o nº de verdadeiros negativos.

Os verdadeiros positivos são aqueles em que a classe + é atribuída correctamente,

[12] Do Inglês, *precision*.
[13] Do Inglês, *recall*.
[14] Do Inglês, *Fscore*.

		Correcta	
		+	−
Prevista	+	VP	FP
	−	FN	VN

Tabela 2.1: Matriz de confusão para as classes positiva (+) e negativa (−).

os falsos positivos são aqueles em que a classe + é atribuída incorrectamente, os falsos negativos são exemplos que não foram classificados como + mas que deveriam ter sido e os verdadeiros negativos são aqueles em que a classe − foi atribuída correctamente.

A precisão é a proporção de exemplos classificados por um sistema e que estão correctos [12]. O valor da precisão é calculado pela expressão

$$\pi = \frac{VP}{VP + FP} \tag{2.7}$$

A cobertura é a proporção de exemplos correctos que são classificados por um sistema [12]. O valor da cobertura é calculado pela expressão

$$\rho = \frac{VP}{VP + FN} \tag{2.8}$$

A medida F_β calcula a média harmónica entre a precisão e a cobertura tornando-se, normalmente, a medida final para comparar sistemas [12] e é dada pela expressão

$$F_\beta = (1 + \beta^2) * \frac{\rho * \pi}{\beta^2 * \rho + \pi} \tag{2.9}$$

Por norma é utilizado o valor $\beta = 1$, obtendo-se a fórmula

$$F_1 = 2 * \frac{\rho * \pi}{\rho + \pi} \tag{2.10}$$

2.4 Aprendizagem de língua natural

Nesta sub-secção apresentam-se os métodos utilizados na aprendizagem de língua natural, quer para marcação de corpora quer para a avaliação de resultados, e apresentam-se os diferentes problemas de língua natural que têm sido abordadas nas tarefas conjuntas das conferências CoNLL [91].

2.4.1 Marcação de corpora

Na aprendizagem de língua natural é necessário criar corpora de acordo com a tarefa a realizar e implica a marcação de segmentos[15]. Os métodos mais frequentes para esta marcação são:

IOB [81]. Neste método cada elemento[16] é etiquetado (juntamente com a devida classe) com uma das etiquetas `I`, `O` ou `B`, para significar dentro[17], fora[18] e início[19], respectivamente. A Figura 2.7 mostra um exemplo da aplicação deste método para a classificação de sintagmas.

Início-Fim [20]**.** Este método é similar ao método IOB uma vez que também classifica os elementos como início, meio e fim alterando apenas a forma de representação: para o início a classe está entre '`(`' e '`*`', o fim da marcação é representado por '`*)`' e os elementos que não indicam o início nem o fim são marcados com '`*`'. A classificação de orações é um do exemplo da aplicação deste método: '`(S*`' para representar o início e '`*)`' para representar o fim.

Este método permite a marcação de termos encaixados; para as orações é possível ter '`(S(S*`' ou '`*))`' e quando existem orações compostas apenas por uma palavra a marcação seria '`(S*S)`' [86]. A Figura 2.8, mostra um exemplo da marcação de orações e de sintagmas com o método Início-Fim.

```
He/B-NP reckons/B-VP the/B-NP current/I-NP account/I-NP
deficit/I-NP will/B-VP narrow/I-VP to/B-PP only/B-np
£/I-np 1.8/I-NP billion/B-NP in/B-PP September/B-NP ./O
```

Figura 2.7: Método IOB: marcação dos sintagmas da frase *'He reckons the current account deficit will narrow to only £ 1.8 billion in September.'*. (Fonte: [85])

2.4.2 Avaliação de resultados

Um sistema é avaliado através da comparação do seu resultado com o resultado de uma marcação manual feita por linguistas e tida como certa [66]. Desta comparação obtêm-se os valores da matriz de confusão (Tabela 2.1).

Na comparação podem ser produzidos 5 tipos de erros [66]:

- o sistema realiza uma marcação onde não devia haver;

[15] Do Inglês, *chunks*.
[16] Do Inglês, *token*.
[17] Do Inglês, *inner*.
[18] Do inglês, *out*.
[19] Do Inglês, *begin*.
[20] Do Inglês, *Start-End*.

```
   *        (S*
(VP*        (S*
   *)         *
(NP*          *
   *)         *)
   *          *
(NP*          *
   *          *
   *)         *
(VP*)         *
(NP*          *
   *)         *
(PP*)         *
(NP*          *
   *          *
   *)         *
   *          *)
```

Figura 2.8: Método Início-Fim: exemplos de marcação de sintagmas e orações. (Fonte: [13])

- o sistema falha uma marcação;
- o sistema acerta na marcação dos limites mas atribui uma classe errada;
- o sistema acerta na classe mas falha a marcação dos limites;
- o sistema atribui a classe errada e falha na marcação dos limites.

A avaliação do desempenho do sistema depende de como estes erros são interpretados e contabilizados. As conferências internacionais adoptaram formas diferente de avaliação [63, 66]:

- nas conferências MUC [39] e HUB-4 [18], um sistema é avaliado em 2 vertentes: a sua habilidade de atribuir a classe correcta e a sua habilidade de acertar nas marcações dos limites no texto. Durante este processo são guardados os valores dos números de respostas correctas, de hipóteses avançadas pelo sistema e o número de possíveis acertos na solução para cada classe. A avaliação final é dada pelo cálculo da media F_1 para todas as classes nas duas vertentes, denominada micro-média da medida F. Todos os 5 tipos de erros, falados anteriormente, são considerados sendo dada uma pontuação parcial a erros que ocorressem apenas numa das vertentes e acertando a outra;
- nas conferências IREX [89] e CoNLL[84, 23, 12, 13]: os sistemas são avaliados e comparados utilizando a micro-média da medida F. Uma marcação está correcta apenas se o sistema atribuir os limites e a classe idênticos à solução marcada por linguistas;
- nas conferências ACE [26] é utilizado um complexo procedimento de avaliação onde estão incluídos mecanismos para abordar várias questões (como por exemplo, marcação parcial de limites ou classe errada). Cada classe possui um

peso parametrizado contribuindo para uma proporção máxima da pontuação final (por exemplo, se uma classe i valer 1 ponto e uma j valer 0.5 então são necessárias 2 classes j para igualar uma classe i na pontuação final do sistema).

2.4.3 As tarefas conjuntas CoNLL

As conferências CoNLL iniciaram-se em 1997 e a partir de 1999 começou a ser incluída uma tarefa conjunta onde são fornecidos, pelos organizadores, dados de treino e teste para permitir a avaliação e comparação dos sistemas participantes de forma sistemática [91]. Durante os últimos 12 anos foram abordados temas como:

- 1999 [68]: reconhecimento de sintagmas nominais em textos;
- 2000 [85]: reconhecimento de todos os sintagmas em textos[21];
- 2001 [86]: reconhecimento de orações em textos;
- 2002 [84] e 2003 [23]: reconhecimento de entidades mencionadas;
- 2004 [12] e 2005 [13]: reconhecimento de papéis semânticos;
- 2006 [9] e 2007 [67]: análise de dependências;
- 2008 [94] e 2009 [42]: temas dos 4 anos anteriores, reconhecimento de papéis semânticos e análise de dependências;
- 2010 [31]: extracção de informação factual de informação incerta ou duvidosa;
- 2011 [74]: modelação de co-referências sem restrições em OntoNotes[22].

[21] Do Inglês, *Chunking*.
[22] Página do projecto: `http://www.bbn.com/ontonotes/`.

Capítulo 3

Extracção de relações entre entidades

Como referido no Capítulo 1, o objectivo final deste trabalho é a extracção de relações entre entidades mencionadas. Esta informação permitirá a navegação inteligente entre documentos, quer através das próprias entidades, quer através das relações existentes entre elas.

A secção 3.1 apresenta a aproximação proposta para extrair relações entre entidades e a secção 3.2 caracteriza o corpus para a língua Portuguesa utilizado. As aplicações desenvolvidas para extrair a informação relativa às entidades e aos argumentos sintácticos são introduzidas na secção 3.3 e a aplicação que interpreta e relaciona as entidades extraídas é apresentada na secção 3.4.

3.1 Aproximação proposta

O problema de identificar num texto as relações existentes entre entidades pode ser dividido em 3 sub-problemas bem definidos:

1. extracção das entidades mencionadas do texto;
2. extracção dos argumentos sintácticos das frases do texto;
3. identificação das relações existentes entre as entidades.

A Figura 3.1 identifica cada um dos sub-problemas e mostra a interacção entre eles:

- Para identificar as entidades presentes no texto foi usado o sistema descrito

em [62, 63]. Este sistema é composto por dois módulos: um etiquetador de categorias gramaticais é responsável pela extracção da informação morfológica das palavras que depois é utilizada pelo identificador de entidades;

- por outro lado, o classificador de argumentos sintácticos extrai essa informação de cada uma das frases do texto;
- as entidades e os argumentos sintácticos são depois utilizados para extrair relações existentes entre as entidades presentes no texto.

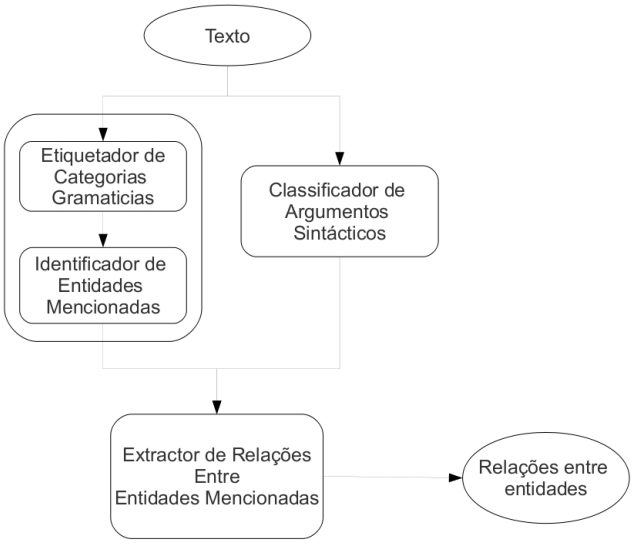

Figura 3.1: Extracção de relações entre entidades: diagrama alto nível.

Qualquer um dos módulos utiliza métodos de aprendizagem supervisionada e como tal necessitam de corpora classificados como dados de treino. Qualquer um dos corpora utilizados (e descritos nos capítulos seguintes) foram construídos com base no Bosque 8.0, um corpus para a língua Portuguesa.

3.2 Bosque 8.0

O Bosque 8.0[1] está incorporado no projecto Floresta Sintá(c)tica e nasceu da necessidade de disponibilizar à comunidade de língua Portuguesa uma ferramenta essencial para ferramentas de processamento de linguagem natural [3].

A Floresta Sintá(c)tica consiste em texto corrido dividido em frases analisadas sintacticamente em estruturas de árvore pelo analisador sintáctico PALAVRAS [5]. A sua revisão foi feita através de uma aproximação utilizando aplicações automáticas e revisão manual [3].

[1] Disponível em `http://www.linguateca.pt/Floresta/corpus.html`.

O Bosque 8.0 é composto por 9368 frases dos primeiros 1000 extractos, totalmente revistos por linguistas, do CETEMPúblico e do CETEMFolha, priorizando a qualidade em detrimento da quantidade [54]. O CETEMPúblico, utiliza Português Europeu e foi criado com notícias do jornal Público [75]; o CETEMFolha utiliza Português do Brasil e foi criado com notícias do jornal Folha de S. Paulo [30] [33].

O Anexo A disponibiliza o glossário das etiquetas e os números presentes no Bosque.

A Figura 3.2 apresenta uma frase do Bosque e a sua análise. Pode-se observar que é uma oração finita (`fcl`), composta por:

- um sintagma nominal (`np`) que é também o sujeito da frase (`SUBJ`). O sintagma é composto por um nome próprio (`prop`);
- um sintagma verbal (`vp`) que é também o predicado da frase (`P`);
- um segundo sintagma nominal (`np`) que constitui um complemento directo (`ACC`).

Ao nível das palavras são indicadas as classes gramaticais (por exemplo, `art` para artigo ou `n` para nome comum) e os respectivos lemas.

```
'source' => 'CP429-7 Vera apagou a luz.',
'number' => 1,
'cod' => 'CETEMPúblico n=429 sec=clt sem=96a',
't' => [
  'fcl||STA',
  [
    'np||SUBJ',
    'prop(\'Vera\' F S)||H::Vera'
  ],
  [
    'vp||P',
    'v-fin(\'apagar\' PS 3S IND)||MV::apagou'
  ],
  [
    'np||ACC',
    'art(\'o\' <artd> F S)||>N::a',
    'n(\'luz\' <np-def> F S)||H::luz'
  ],
  'jjpunct(-.-)'
]
```

Figura 3.2: Bosque 8.0: representação da frase '`Vera apagou a luz.`'.

3.3 Extracção de informação

Para conseguir relacionar as entidades existentes num texto começou-se por extrair as entidades do texto e os argumentos sintácticos das frases. As sub-secções seguintes descrevem estas ferramentas.

3.3.1 Extracção de entidades mencionadas

O reconhecimento de entidades é a tarefa de identificar e classificar elementos existentes num texto em determinadas categorias como nomes de pessoas, locais e organizações [62]. Após a classificação é possível determinar o assunto geral de um documento com base nas entidades ou até mesmo aglomerar documentos que possuam entidades em comum [62, 63].

As abordagens a este problema específico são diversas, desde baseadas em gramáticas de regras [1] até sistemas baseados em aprendizagem automática [64] tal como enunciado em [62].

A Figura 3.3 ilustra a aplicação usada para reconhecer as entidades [62]. Esta aplicação é formada por dois blocos: o primeiro extrai informação morfológica e atributos ortográficos dos elementos do texto que é utilizada pelo segundo para identificar e classificar as entidades.

O Reconhecedor de Entidades utiliza o LibSVM[2] [15], uma implementação de máquinas de vectores de suporte. O modelo criado utiliza um corpus composto por notícias dos jornais Oje [92], Record [29] e Público[75] onde as entidades foram marcadas manualmente ao qual foram adicionados atributos ortográficos e morfológicos das palavras.

Os atributos morfológicos são obtidos através de um etiquetador de categorias gramaticais desenvolvido no âmbito desta dissertação, e que será descrito em pormenor no Capítulo 4. A descrição do Reconhecedor de Entidades pode ser consultada em [62, 63].

3.3.2 Extracção de argumentos sintácticos

A extracção de argumentos sintácticos de uma frase permite obter informação sobre as funções sintácticas dos segmentos de uma frase. Estas funções incluem a acção (predicado da frase), quem realiza a acção (sujeito da frase) e ainda os seus complementos (directo, indirecto, etc.) [13].

Actualmente é um dos subgrupos mais activos na área de processamento de linguagem natural, tendo a maior parte da sua pesquisa realizada a partir de 2004 [13].

[2]Disponível em `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

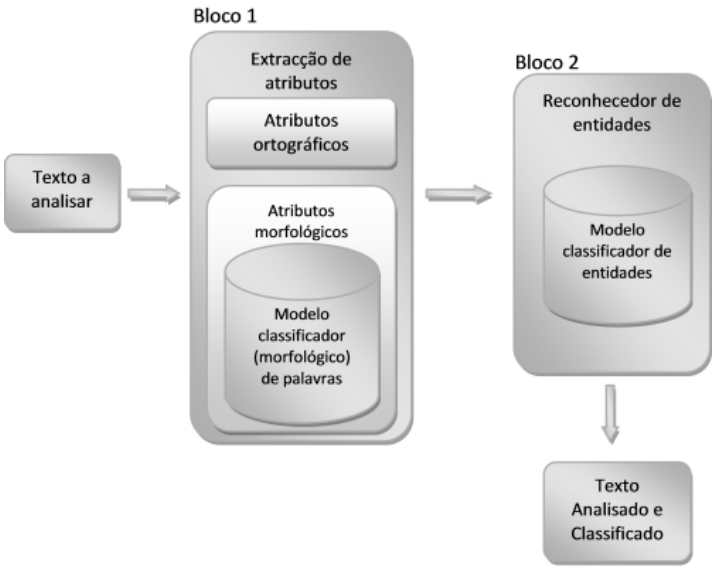

Figura 3.3: Reconhecedor de entidades: diagrama de blocos.(Fonte: [62])

As abordagens a este problema são equivalentes às utilizadas na extracção de entidades mencionadas em 3.3.1, quer através do desenvolvimento de gramáticas, quer utlizando técnicas aprendizagem automática [36].

No âmbito deste trabalho, utilizaram-se técnicas de aprendizagem automática para encontrar um modelo preliminar para o extractor de argumentos sintácticos. Foi utilizado o MinorThird[3], uma ferramenta que implementa diversos métodos para extracção e classificação de *spans* tais como os campos condicionais aleatórios [105] e os modelos condicionais de Markov [93].

O classificador de argumentos sintácticos desenvolvido no âmbito desta dissertação será descrito e analisado no Capítulo 5.

3.4 Identificação de relações entre entidades

Com a informação das entidades e dos argumentos sintácticos torna-se possível relacioná-las. Para tal foi implementada uma aplicação que, fazendo uma análise frase a frase, utiliza a informação obtida pelo reconhecedor de entidades e o classificador de argumentos sintácticos para extrair as relações.

A Figura 3.4 mostra o resultado obtido para a frase "A PT vendeu a Vivo.". Na interface da aplicação os argumentos sintácticos são marcados utilizando um sistema de cores: verde para o `Sujeito`; rosa para o `Predicado`; azul para o `Complemento Directo`. Enquanto que as entidades são marcadas a negrito.

[3]Disponível em `http://sourceforge.net/apps/trac/minorthird/wiki`.

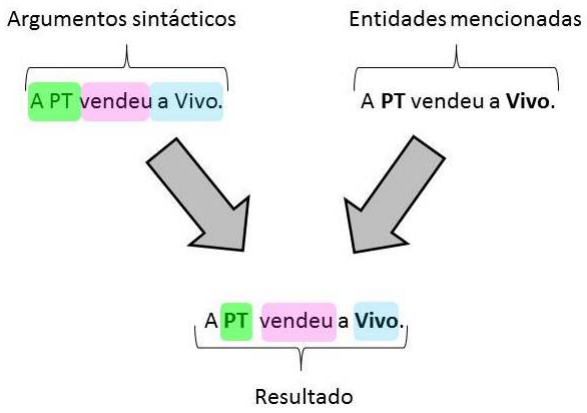

Figura 3.4: Extractor de relações entre entidades: exemplo de aplicação.

Como seria de esperar, a utilização desta aplicação necessita de dois ficheiros de marcação (sobre o mesmo texto): um com as entidades e outro com os argumentos sintácticos.

A aplicação permite ainda a escrita das relações identificadas para um ficheiro. Para cada frase processada é incluída a própria frase, as entidades presentes na frase e os argumentos sintácticos `Agente`, `Acção` e `Objecto`. A Figura 3.5 mostra um exemplo de uma frase com a identificação das relações existentes entre entidades.

```
Frase: A PT vendeu a Vivo.
Entidades: PT; Vivo

Agente: PT
Acção: vender
Objecto: Vivo
```

Figura 3.5: Informação extraída da frase `'A PT vendeu a Vivo.'`.

Capítulo 4

Etiquetador de categorias gramaticais

O etiquetador de categorias gramaticais[1] descrito neste capítulo foi implementado pela necessidade de utilização de informação gramatical na construção do sistema reconhecedor de entidades descrito em [62, 63].

Para o seu desenvolvimento foi necessário construir um corpus anotado utilizando informação do corpus Bosque 8.0 [3] (ver secção 3.2) e do léxico Label-Lex [50]. A este corpus chamou-se POS-Publico.

A secção 4.1 define a tarefa de etiquetar as palavras com a sua categoria gramatical, e a secção 4.2 apresenta vários etiquetadores desenvolvidos para a língua Portuguesa e descreve as ferramentas FreeLing e SVMTool que permitem criar etiquetadores a partir de corpora anotados. A secção 4.3 descreve a construção do corpus POS-Publico. Finalmente, a secção 4.4 apresenta e discute os resultados obtidos com o etiquetador construído.

4.1 A tarefa

Em muitas aplicações de processamento de linguagem natural é necessário utilizar informação sobre as categorias gramaticais das palavras presentes no texto e a função do etiquetador é atribuir uma etiqueta gramatical a cada palavra. As etiquetas utilizadas no Bosque 8.0 podem ser consultadas no Anexo A.

[1] Do Inglês, *part-of-speech tagger*.

A Figura 4.1 ilustra um exemplo de uma frase classificada gramaticalmente; é possível observar que existe um nome próprio (`Vera`), um verbo (`apagou`), um determinante (`a`), um nome (`luz`) e um elemento de pontuação (`.`).

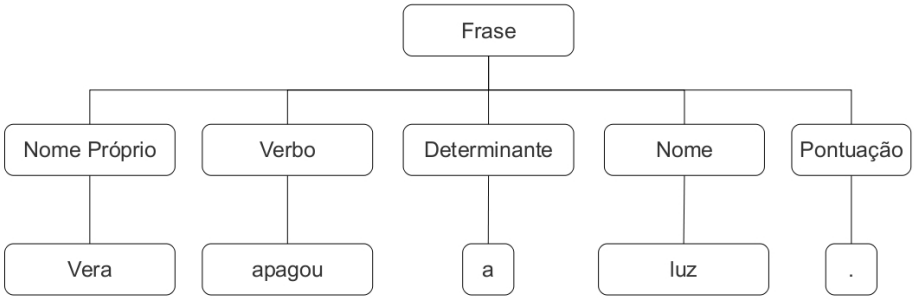

Figura 4.1: Classificação gramatical da frase `'Vera apagou a luz.'`.

Segundo Voutilainen [103] a tarefa de classificação de categorias gramaticais pode ser realizada em dois passos:

1. etiquetação introduzindo ambiguidade, ou seja, etiquetar cada palavra com todas as categorias possíveis;
2. desambiguação eliminando as alternativas improváveis.

As abordagens mais proeminentes para a tarefa de classificação de categorias gramaticais são a linguística e a orientada a dados. Enquanto a primeira utiliza abstracções sobre os paradigmas e sintagmas da linguagem criando gramáticas de regras para descartar alternativas improváveis aquando da classificação, a abordagem orientada a dados utiliza corpora classificados como treino.

Segundo Dickinson [24], nos 10 anos anteriores a 2006 a evolução do desempenho de dos etiquetadores de categorias gramaticais não sofreu muitas alterações, existindo mesmo a possibilidade ter sido atingido um limite superior no desempenho.

Em 1996 um desempenho estado da arte era considerado por um valor de exactidão de 96.63% [82], enquanto que 10 anos depois um desempenho estado da arte reportava valores de 97.24% [99], indicando um aperfeiçoamento inferior a 1% [24]. Actualmente valores perto nos 97% são considerados bons desempenhos.

4.2 Trabalho relacionado

Nesta tarefa de processamento de língua natural existe uma diversidade de ferramentas disponíveis para a língua Portuguesa. A Linguateca [53] mantém, na página apropriada, ligações para muitas delas.

Entre os exemplos apresentados, e para o Português do Brasil, estão o classificador Aelius[2] e o CURUPIRA[3].

Para o Português Europeu está disponível um etiquetador implementado com base no *Tree-Tagger* [87, 88] por Pablo Gamallo[4] da Universidade de Santiago de Compostela que também pode ser usado para a língua Galega e o classificador LX-Suite[5] [7] implementado pelo grupo NLX (*Natural Language and Speech Group*) da Universidade de Lisboa. Este etiquetador utiliza a ferramenta *Tnt* [8] para gerar o modelo e obteve uma precisão de 96.87% com validação cruzada 10 pastas sobre o corpus LX-Corpus [7].

O FreeLing e o SVMTool são ferramentas que permitem criar etiquetadores de categorias gramaticais a partir de corpora anotados. Estas ferramentas são abordadas nas sub-secções seguintes.

4.2.1 FreeLing

O FreeLing[6] é uma biblioteca de análise linguística implementada no centro de aplicações e tecnologias para linguagem e discurso (TALP) da Universidade Politécnica da Catalunha. Os idiomas suportados são o Inglês, Espanhol, Catalão, Português, Italiano, Galego, Galês e Asturiano, sendo o corpus da língua Portuguesa baseado no Bosque [69].

Esta biblioteca pode ser usada para realizar análises morfológicas, reconhecimento de datas e números e classificação de categorias gramaticais [69]. Os analisadores existentes no FreeLing são baseados em gramáticas livres de contexto[7] [47] [69].

O FreeLing gera vários tipos de saída como análise morfológica das palavras ou a análise de dependências[8]. A análise da estrutura de dependências da frase é obtida através de meios automáticos ou de gramáticas [69].

Ao utilizar o FreeLing, verificaram-se as seguintes características:

- o corpus utilizado está pré-processado na aplicação. Na fase de classificação, faz a partição de palavras compostas nos seus constituintes básicos e realiza junção (por *underscores*) de palavras que são nomes próprios ou frases preposicionais;

- o conjunto de etiquetas não era o desejável. O conjunto utilizado no FreeLing

[2]Disponível em `http://aelius.sourceforge.net/`.
[3]Disponível em `http://www.nilc.icmc.usp.br/nilc/tools/curupira.html`.
[4]Disponível em `http://gramatica.usc.es/~gamallo/tagger.htm`.
[5]Disponível em `http://lxcenter.di.fc.ul.pt/services/pt/LXServicesSuitePT.html`.
[6]Disponível em `http://nlp.lsi.upc.edu/freeling/`.
[7]Do Inglês, *context-free grammar* (CFG).
[8]Do Inglês, *dependency parsing*.

é denominado por EAGLES[9]. Este conjunto era muito específico nas etiquetas (como por exemplo aborda os diferentes tempos verbais, número, género entre outros), variando o tamanho de cada com base na palavra abordada (por exemplo, `NCFS000` para nome comum singular feminino ou `VMN0000` para um verbo principal no infinitivo) e não podia ser alterado porque já está embutido no corpus treinado no FreeLing.

- o processamento era mais lento que o realizado pela ferramenta SVMTool (experiências realizadas para a língua Inglesa).

4.2.2 SVMTool

O SVMTool[10] é um gerador de analisadores sequenciais que utiliza máquinas de vectores de suporte [101] também implementado no centro de aplicações e tecnologias para linguagem e discurso (TALP) da Universidade Politécnica da Catalunha. As suas principais características são [38]:

- simplicidade, devido à sua fácil configuração;
- flexibilidade, devido à fácil extracção de características dos dados de entrada;
- portabilidade, devido a ser independente do idioma, tendo sido aplicado com sucesso à língua Inglesa e Espanhola;
- precisão elevada com valores acima dos 97% no corpus do *Wall Street Journal*;
- eficiência que depende do tamanho do texto a processar e da versão utilizada. (das versões disponíveis a implementada em C++ é a mais rápida mas foi descontinuada pelos autores; este trabalho utiliza a implementação em Perl);
- robustez porque possui estratégias para tratar o ruído e palavras desconhecidas.

A aplicação possui três componentes:

- um gerador de modelos, SVMTLearn, que recebe um corpus num formato específico, extrai a informação necessária para criar os vectores de características e cria um modelo utilizando o SVM^{light} [44], uma implementação de máquinas de vectores de suporte;
- um classificador, SVMTagger, que recebe o texto a classificar e um modelo (criado pelo SVMTLearn) e indica, para cada palavra do texto original a sua categoria gramatical;

[9] Página do projecto em: `http://www.ilc.cnr.it/EAGLES/annotate/annotate.html`.
[10] Disponível em `http://www.lsi.upc.edu/~nlp/SVMTool/`.

- um avaliador, SVMTeval, que recebe a saída do SVMTagger, compara-o com um ficheiro padrão dourado[11], ficheiro com a categoria correcta de cada palavra, e calcula os valores da precisão, cobertura e de F_1 para cada classe.

Ao utilizar o SVMTool, verificaram-se as seguintes características:

- a possibilidade de usar um corpus pensado e construído com características desejadas;
- a velocidade da classificação;
- a inexistência de pós-processamento dos textos, após a classificação.

4.3 Construção do corpus POS-Publico

A escolha da utilização do SVMTool gerou a necessidade de criar um corpus anotado com categorias gramaticais devido a algumas características específicas das palavras presentes no Bosque 8.0: as palavras pertencentes a nomes próprios são aglutinadas por *underscores* e é feita a expansão das contracções da língua. Estas características, que podem ser observadas na Figura 4.2, impossibilitam a utilização directa de novos textos.

Desta forma, um novo corpus, designado **POS-Publico**, foi criado da seguinte forma:

1. utilizou-se o ficheiro `CETEMPúblico`, que tem frases escritas em Português Europeu;
2. consideraram-se apenas as frases terminadas com sinais de pontuação;
3. fez-se a reposição da contracção das palavras de acordo com a Tabela 4.1 e criaram-se novas etiquetas gramaticais para estas palavras. Por exemplo a contracção '`da`' é constituída por uma preposição e um artigo definido, tendo sido atribuída a etiqueta `prp-artdef`. As etiquetas utilizadas estão representadas na Tabela 4.2;
4. adoptaram-se as etiquetas do Label-Lex (Anexo B). Não foram mantidas as etiquetas do Bosque porque têm muita especificidade (como por exemplo, `conj-c` para conjunções coordenativas, `conj-s` para subordinativas e 4 etiquetas para verbos), existem casos como `n-adj` em que as palavras podiam ser tanto nomes como adjectivos e as contracções não têm etiqueta associada. A Tabela 4.3 mostra o mapeamento entre as etiquetas do Bosque e as do corpus construído que foi realizado em 3 passos intermédios:

[11] Do Inglês, *gold standard*.

```
'source' => 'CP1-2 O 7 e Meio é um ex-libris da noite algarvia.',
'number' => 1,
'cod' => 'CETEMPúblico n=1 sec=clt sem=92b',
't' => [
  'fcl||STA',
  [
    'np||SUBJ',
    'art(\'o\' <artd> M S)||>N::O',
    'prop(\'7_e_Meio\' M S)||H::7_e_Meio'
  ],
  [
    'vp||P',
    'v-fin(\'ser\' PR 3S IND)||MV::é'
  ],
  [
    'np||SC',
    'art(\'um\' <arti> M S)||>N::um',
    'ec(\'ex-\')||>N::ex-',
    'n(\'libris\' M P)||H::libris',
    [
      'pp||N<',
      'prp(\'de\' <sam->)||H::de',
      [
        'np||P<',
        'art(\'o\' <-sam> <artd> S)||>N::a',
        'n(\'noite\' <np-def> F S)||H::noite',
        [
          'adjp||N<',
          'adj(\'algarvio\' F S)||H::algarvia'
        ]
      ]
    ]
  ],
  'jjpunct(-.-)'
]
```

Figura 4.2: Bosque 8.0: representação da frase 'O 7 e Meio é um ex-libris da noite algarvia.'.

(a) partição de grupos de palavras contendo *underscores* ('_');

(b) atribuição das etiquetas do Label-Lex, por conversão directa. Para os números ou nomes próprios (que não existem no Label-Lex) foram criadas e atribuídas as respectivas etiquetas;

(c) as palavras não existentes no Label-Lex ou que possuíam mais do que uma categoria foram etiquetadas como `Ambígua`.

5. as palavras etiquetadas como `Ambígua` foram revistas manualmente e identificada a etiqueta correcta;

6. finalmente, procuraram-se as palavras terminadas em *hífen*, para proceder à sua separação: a palavra fica com a categoria já identificada e o *hífen* foi classificado como sinal de pontuação. Um excerto do corpus POS-Publico pode ser observado na Figura 4.3.

```
O                       DET
7                       NUM
e                       CONJ
Meio                    PROP
é                       V
um                      DET
ex                      PFX
-                       PU
libris                  N
da                      PREPXDET
noite                   N
algarvia                ADJ
.                       PU
```

Figura 4.3: POS-Publico: excerto do corpus para a frase '`O 7 e Meio é um ex-libris da noite algarvia.`'.

4.4 O etiquetador

A construção do corpus descrita na secção anterior permitiu a criação de um modelo SVMTool para a língua Portuguesa. As sub-secções seguintes descrevem a configuração experimental do etiquetador e o desempenho obtido.

4.4.1 Configuração experimental

Para avaliar o desempenho do classificador utilizou-se uma divisão treino-teste com dois terços para treino (2951 frases e 88692 palavras) e um terço para teste (1476 frases e 44977 palavras). A Tabela 4.4 indica o nº de palavras de cada etiqueta para o corpus e os conjuntos de treino e teste.

Palavras	**Palavras Principais**			
Depend.	**De/de**	**Em/em**	**A/a**	**Por/por**
o	Do/do	No/no	Ao/ao	Pelo/pelo
a	Da/da	Na/na	À/à	Pela/pela
os	Dos/dos	Nos/nos	Aos/aos	Pelos/pelos
as	Das/das	Nas/nas	Às/às	Pelas/pelas
um	Dum/dum	Num/num		
uma	Duma/duma	Numa/numa		
uns	Duns/duns	Nuns/nuns		
umas	Dumas/dumas	Numas/numas		
ele	Dele/dele	Nele/nele		
ela	Dela/dela	Nela/nela		
eles	Deles/deles	Neles/neles		
elas	Delas/delas	Nelas/nelas		
este	Deste/deste	Neste/neste		
esta	Desta/desta	Nesta/nesta		
estes	Destes/destes	Nestes/nestes		
estas	Destas/destas	Nestas/nestas		
aquele	Daquele/daquele	Naquele/naquele	Àquele/àquele	
aquela	Daquela/daquela	Naquela/naquela	Àquela/àquela	
aqueles	Daqueles/ daqueles	Naqueles/ naqueles	Àqueles/ àqueles	
aquelas	Daquelas/ daquelas	Naquelas/ naquelas	Àquelas/ àquelas	
esse	Desse/desse	Nesse/nesse		
essa	Dessa/dessa	Nessa/nessa		
esses	Desses/desses	Nesses/nesses		
essas	Dessas/dessas	Nessas/nessas		
aí	Daí/daí			
ali	Dali/dali			
aqui	Daqui/daqui			
aquilo	Daquilo/daquilo	Naquilo/naquilo		
isso	Disso/disso	Nisso/nisso		
isto	Disto/disto	Nisto/nisto		
outro		Noutro/noutro		
outra		Noutra/noutra		
outros		Noutros/noutros		
outras		Noutras/noutras		
onde			Aonde/aonde	

Tabela 4.1: Contracções realizadas na criação do corpus (acontecem quando no Bosque aparecem as palavras principais seguidas das dependentes).

prp-prondet			
Do/do	Da/da	Dos/dos	Das/das
No/no	Na/na	Nos/nos	Nas/nas
Ao/ao	À/à	Aos/aos	Às/às
Pelo/pelo	Pela/pela	Pelos/pelos	Pelas/pelas
prp-artindef			
Dum/dum	Duma/duma	Duns/duns	Dumas/dumas
Num/num	Numa/numa	Nuns/nuns	Numas/numas
prp-adv			
Daí/daí	Dali/dali	Daqui/daqui	Aonde/aonde
prp-prondet			
Dele/dele	Daquilo/daquilo	Dela/dela	Disso/disso
Disto/disto	Delas/delas	Nele/nele	Deste/deste
Desta/desta	Neles/neles	Destes/destes	Nelas/nelas
Neste/neste	Daquele/daquele	Nesta/nesta	Daquela/daquela
Daqueles/daqueles	Nestas/nestas	Daquelas/daquelas	Naquele/naquele
Naquela/naquela	Dessa/dessa	Naqueles/naqueles	Desses/desses
Dessas/dessas	Nesse/nesse	Nessa/nessa	Nesses/nesses
Naquilo/naquilo	Nisto/nisto	Noutro/noutro	Noutra/noutra
Noutras/noutras	Àquele/àquele	Àquela/àquela	Àqueles/àqueles
Nisto/nisto	Nela/nela	Destas/destas	Nestes/nestes
Deles/deles	Desse/desse	Naquelas/naquelas	Nessas/nessas
Noutros/noutros	Àquelas/àquelas		

Tabela 4.2: Etiquetas gramaticais atribuídas às contracções.

Na criação do modelo utilizaram-se os parâmetros por omissão do SVMTool:

- janela de tamanho 5 com a palavra a treinar na posição central;
- as palavras que apareciam menos de duas vezes em cem mil palavras não foram consideradas;
- validação cruzada 10 pastas;
- direcção do treino foi da esquerda para a direita e vice-versa.

O desempenho do modelo foi medido através da precisão (π), cobertura (ρ) e medida F_1.

4.4.2 Resultados obtidos

A Tabela 4.5 apresenta os valores de precisão, cobertura e F_1 obtidas no conjunto de teste para cada uma das categorias gramaticais.

Da observação da Tabela 4.5 é possível verificar que:

- as classes com melhores valores de precisão foram a `PREPXPRO`, `PREPXADV`, `PREPXDET` e a `PU`;
- as classes com piores valores de precisão foram a `ADJ`, `NPT` e `INTERJ`;

	Iniciais	**Finais**	
Categorias	**Bosque**	**Label-Lex**	**Criadas**
Substantivo/nome	n	N	
Substantivo/adjectivo	n-adj	Classificadas noutras categorias	
Adjectivo	adj	ADJ	
Nome Próprio	prop		PROP
Advérbio	adv	ADV	
Verbo finito	v-fin	V	
Verbo gerúndio	v-ger	V	
Verbo particípio	v-pcp	V	
Verbo infinitivo	v-inf	V	
Pontuação	pu		PU
Artigos	art	DET	
Pronome determinativo	pron-det	PRO	
Pronome independente	pron-indp	PRO	
Pronome pessoal	pron-pers	PRO	
Preposição	prp	PREP	
Interjeição	intj	INTERJ	
Conjunção subordinativa	conj-s	CONJ	
Conjunção coordenativa	conj-c	CONJ	
Prefixo	ec	PFX	
Número	num		NUM
Palavra estrangeira	x		NPT
Contracções			
Preposição + artigo definido	prp-artdef	PREPXDET	
Preposição + artigo indefinido	prp-artindef	PREPXDET	
Preposição + pronome determinativo	prp-prondet	PREPXPRO	
Preposição + advérbio	prp-adv	PREPXADV	

Tabela 4.3: Mapeamento entre as etiquetas do Bosque e as utilizadas no novo corpus.

Etiqueta	**%**	**Corpus**	**Treino**	**Teste**
N	18.4	24657	16347	8310
PROP	8.5	11346	7515	3831
ADJ	5.0	6739	4505	2234
V	12.2	16244	10740	5504
DET	7.6	10102	6719	3383
PRO	5.3	7081	4705	2376
ADV	4.6	6152	4050	2102
PREP	9.4	12597	8348	4249
INTERJ	0.0	13	10	3
CONJ	3.8	5115	3388	1727
PU	15.6	20856	13852	7004
PREPXDET	7.5	9997	6648	3349
PREPXPRO	0.1	175	120	55
PREPXADV	0.0	3	2	1
PFX	0.1	160	110	50
NUM	1.7	2270	1521	749
NPT	0.1	162	112	50

Tabela 4.4: Proporção e nº de palavras de cada etiqueta: o corpus e conjuntos de treino e teste.

Etiqueta	π	ρ	F_1
N	0.965	0.964	0.965
PROP	0.986	0.991	0.988
ADJ	0.886	0.891	0.888
V	0.973	0.976	0.974
DET	0.960	0.979	0.969
PRO	0.930	0.934	0.932
ADV	0.938	0.951	0.944
PREP	0.991	0.977	0.984
INTERJ	0	0	0
CONJ	0.967	0.932	0.949
PU	0.999	1	0.999
PREPXDET	0.999	1	0.999
PREPXPRO	1	0.964	0.982
PREPXADV	1	1	1
PFX	0.935	0.860	0.896
NUM	0.976	0.955	0.963
NPT	0.872	0.739	0.800

Tabela 4.5: Desempenho do etiquetador de categorias gramaticais.

- as classes com melhores valores de cobertura foram a `PU`, `PREPXDET` e `PREPXADV`;
- as classes com piores valores de cobertura foram a `PFX`, `NPT` e `INTERJ`;
- as classes com melhores valores de F_1 foram a `PREPXADV`, `PREPXDET` e `PU`;
- as classes com piores valores de F_1 foram a `ADJ`, `NPT` e `INTERJ`;

Observando a Tabela 4.4 conclui-se que as categorias que estão nos limites inferiores e superiores da precisão, cobertura e F_1 possuem número reduzido de exemplos o que explica estas anormalidades nos resultados. As restantes categorias obtiveram valores de precisão acima de 87%, de cobertura acima de 73% e de F_1 acima dos 80%.

A Tabela 4.6 apresenta a média e desvio padrão para o conjunto de categorias: a precisão média é superior a 95% e a cobertura média é superior a 94%. Para esta média não foram consideradas as categorias `INTERJ` e `PREPXADV` por possuírem um número de exemplos muito baixo.

Desempenho	π	ρ	F_1
Média	0.958	0.941	0.949
Desvio Padrão	0.040	0.068	0.053

Tabela 4.6: Valores médios de desempenho do classificador (exclui-se as categorias `INTERJ` e `PREPXADV`.

A precisão deste etiquetador quando comparada com a precisão obtida pelo LX-Suite é inferior em 1.07%. Quando comparado com os valores de precisão obtidos para a língua Inglesa em [38], este etiquetador possui uma precisão inferior em 1.4%.

Por fim, é possível constatar que estes valores estão abaixo dos valores referidos em [24], mas sendo a diferença pequena e tratando-se de um etiquetador para outra língua, pode-se considerar que o etiquetador implementado possui um desempenho equivalente aos referidos em outros projectos nacionais e internacionais.

Capítulo 5

Classificador de argumentos sintácticos

Como indicado no Capítulo 3, o objectivo do classificador de argumentos sintácticos descrito neste capítulo é extrair relações entre entidades encontradas em textos escritos em língua Portuguesa.

Tendo em conta esta tarefa genérica foi desenvolvido um corpus anotado que reúne informação de diversas fontes com o mesmo formato do utilizado nas tarefas conjuntas do CoNLL de 2004 [12] e de 2005 [13].

A secção 5.1 define a tarefa de classificação de argumentos sintácticos e a secção 5.2 apresenta os trabalhos desenvolvidos quer a nível internacional quer para a língua Portuguesa. A secção 5.3 descreve os corpora construídos para esta tarefa. Finalmente, a secção 5.4 introduz o sistema preliminar construído descrevendo a ferramenta e os corpora utilizados; apresenta depois a configuração experimental e os resultados obtidos.

5.1 A tarefa

A classificação de argumentos sintácticos é actualmente um dos subgrupos mais activos na área de processamento de linguagem natural. Consiste em identificar os verbos presentes numa frase e os seus argumentos sintácticos [13] tais como o sujeito e o objecto da acção (entre outros).

A Figura 5.1 apresenta uma frase disponível no Bosque 8.0 [3]. A análise da frase

permite verificar que possui um sujeito (`Vera`), um verbo que compõe o predicado (`apagou`) e um complemento directo que sofre a acção realizada pelo sujeito (`a luz`).

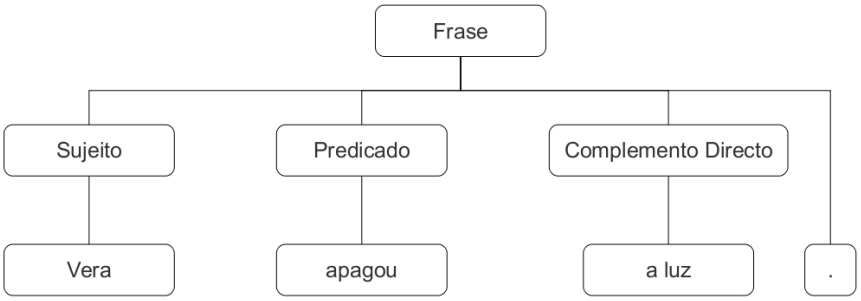

Figura 5.1: Argumentos sintácticos da frase `'Vera apagou a luz.'`.

Gildea e Jurafsky [36], pioneiros na classificação de argumentos sintácticos, enumeram dois métodos proeminentes para realizar a análise de textos, um baseado em gramáticas[1] e outro orientado a dados.

O processo de criar gramáticas é moroso visto serem criadas à mão e necessitarem de incluir uma descrição para cada disposição, existente na língua, dos argumentos em relação aos verbos [36].

Para os sistemas orientados a dados são necessários corpora classificados, e aplicações capazes de criar modelos a partir dos mesmos. Esses modelos são posteriormente usados para classificar textos sem marcações.

A próxima secção apresenta exemplos de aplicações que realizam a classificação de argumentos sintácticos utilizando estas diferentes abordagens.

5.2 Trabalho relacionado

Ao longo da pesquisa efectuada foi possível verificar que, para a língua Portuguesa, esta ainda é uma área pouco explorada. A única aplicação concreta encontrada foi desenvolvida por Bick [6] e será analisada na secção seguinte.

5.2.1 Sistemas baseados em gramáticas

O sistema desenvolvido por Bick [6] utiliza uma gramática de restrições para mapear e desambiguar 40 tipos diferentes de argumentos sintácticos. A gramática criada possui 500 regras de mapeamento, um pequeno número de regras de desam-

[1] Do Inglês, *grammar based systems*.

biguação e obteve um valor médio de F_1 de 88.6%. Como já referido, este sistema foi desenvolvido para a língua Portuguesa.

Gildea e Hockenmaier [35] utilizam um classificador estatístico baseado numa gramática combinatória de categorias[2] (CCG). O classificador apresentado é treinado e testado num corpus de derivações obtidas da conversão das estruturas presentes no *Penn Treebank* e comparado com um classificador baseado no de Collins [20]. Os resultados obtidos permitiram verificar que quando são comparados todos os argumentos ambos os classificadores possuem um desempenho similar (na medida F_1 o classificador de Collins obteve 66.4% e o de CCG 66.8%) mas quando são comparados os argumentos do núcleo (`A0...A5`) o CCG obteve um desempenho superior com um valor de F_1 igual a 74.8% contra 72.6% para o de Collins.

Roy [83] abordada as estratégias de 3 classificadores baseados em regras usados para extrair informação semântica sobre os argumentos:

- Analisador Link [98]. A informação dos argumentos é extraída através das dependências existentes entre as palavras dos textos. A gramática usada possui um conjunto de palavras como símbolos terminais, um conjunto de relações entre duas palavras e um conjunto de propriedades usadas como requerimentos das relações;

- Minipar [52]. É um classificador da língua Inglesa baseado em princípios. A gramática é representada por um rede de nós onde cada é um tipo de categoria gramatical e as ligações representam tipos de relações sintácticas;

- XLE [21]. Utiliza uma gramática funcional lexical[3] (LFG). A LFG assenta principalmente em duas estruturas: (i) estrutura do constituinte onde são representadas as palavras e a estrutura da frase; (ii) estrutura funcional onde são representadas as funções gramaticais.

Zhang [109] utiliza uma gramática para realizar a classificação de argumentos sintácticos com os dados usados na tarefa conjunta do CoNLL 2005 e reporta F_1 de 78.13%. O sistema realiza um pré-processamento com o analisador de Charniak [16] e possui 4 fases [109]:

1. Poda das árvores[4] obtidas pelo analisador de Charniak. Por meio de heurísticas introduzidas em [108] são filtrados os elementos que não são argumentos dos predicados;

2. Identificação de argumentos. Realizada por um classificador binário com base em máquinas de vectores de suporte;

[2] Do Inglês *Combinatory Categorial Grammar*.
[3] Do Inglês *Lexical Functional Grammar*.
[4] Do inglês *pruning*.

3. Classificação de argumentos. Realizada por um classificador criado com o SVM^{light} [44]. O *Tree Kernel* feito por Moschitti [65] sobre o SVM^{light} foi alterado para ser baseado em gramáticas;
4. Pós-processamento baseado em regras introduzidas em [55]. Nesta fase são feitas correcções tais como tratar argumentos sobrepostos e argumentos que possam ter os limites mal marcados.

Existem aproximações cujo objectivo é classificar os predicados, elementos que estão na base da posterior tarefa da classificação de argumentos. Duran *et. al.* [28] e Hendrickx *et. al.* [43] abordam o tema dos predicados complexos, para Português do Brasil e Europeu respectivamente.

Os predicados complexos são predicados que possuem mais do que uma palavra; são construções em que cada componente possui parte da informação associada a um predicado simples. Hendrickx *et. al.*[43] referem a anotação de 2000 predicados complexos no corpus CINTIL[5] [4] para predicados complexos compostos só por verbos ou compostos por verbos e nomes.

O estudo da classificação de predicados complexos fornece recursos cada vez mais precisos para futuras abordagens de processamento de linguagem natural [28].

5.2.2 Sistemas orientados a dados

A maioria dos sistemas orientados a dados foram desenvolvidos no âmbito das tarefas conjuntas das conferências CoNLL. Nas edições de 2004 [12] e 2005 [13] foi introduzido o tema da classificação de argumentos sintácticos; nas edições de 2008 [94] e 2009 [42] também foi abordada a classificação de argumentos sintácticos mas os corpora, para além da informação usada nas edições de 2004 e 2005, possuíam informação sobre análise de dependências [47]. A análise de dependências foi abordada nas edições de 2006 [9] e 2007 [67].

A Conferência de Aprendizagem Computacional de Língua Natural[6] (CoNLL) é realizada todos os anos pelo Grupo de Interesse Especial em Aprendizagem de Língua Natural da Associação de Linguística Computacional[7] (SIGNLL) desde 1997. Todos os anos é escolhido um tema sobre o qual são realizadas palestras e desde 1999 é realizada uma tarefa conjunta com uma componente prática, onde participam vários sistemas com diferentes abordagens ao tema [91].

De seguida são analisadas as tarefas conjuntas das edições de 2004 e 2005 desta conferência. Optou-se por estas edições visto constituírem uma primeira abordagem ao problema e utilizarem um tipo de informação similar àquela utilizada no corpus construído no âmbito desta dissertação.

[5] Disponível em `http://cintil.ul.pt/`.
[6] Do Inglês, *Conference on Computational Natural Language Learning.*
[7] Do Inglês, *Special Interest Group on Natural Language Learning.*

Tarefa conjunta do CoNLL 2004. Este estudo, dedicado à língua Inglesa, utiliza o corpus *Penn Treebank* [58, 59] ao qual foram posteriormente acrescentadas as estruturas predicado-argumento presentes no *PropBank* [46, 70].

A avaliação dos resultados é realizada pelos valores de precisão, cobertura e medida F_1. Um argumento é classificado correctamente quando todas as palavras de um *span*[8] são reconhecidas e a classificação é correcta. Para a avaliação dos sistemas não são considerados os verbos pois são fornecidos com os dados de teste [12].

O corpus é constituído por seis secções do *Wall Street Journal* [27]: as secções 15 a 18 (8936 frases) são utilizadas para treino, a secção 20 (1671 frases) para desenvolvimento e secção 21 (2012 frases) para teste.

As classes possíveis para os argumentos sintácticos são: argumentos numerados, adjuntos, referências e verbos. Os argumentos numerados possuem a nomenclatura `A0` – `A5` e `AA`, sendo `A0` o sujeito, também denominado por agente; o `A1` o objecto, também denominado por paciente ou tema; os restantes são argumentos específicos de cada verbo podendo por isso variar na sua presença ou não [46, 70]; os adjuntos são argumentos opcionais que podem existir ou não numa frase; as referências são argumentos que representam argumentos enumerados noutra parte da frase [12].

A Figura 5.2 ilustra os dados de treino fornecidos para a tarefa. Cada linha possui informação relativa a uma palavra da frase e inclui: a palavra e sua categoria gramatical, os sintagmas no formato IOB, as orações no formato Inicio-Fim, as entidades mencionadas no formato IOB (Pessoa, Local, Organização e Outros), os verbos alvo e respectivos argumentos sintácticos no formato Início-Fim (também conhecidos como molduras semânticas[9] presentes no *PropBank*.

```
The          DT   B-NP    (S*       O      -          (A0*              *
San          NNP  I-NP      *       B-ORG  -            *               *
Francisco    NNP  I-NP      *       I-ORG  -            *               *
Examiner     NNP  I-NP      *       I-ORG  -            *A0)            *
issued       VBD  B-VP      *       O      issue      (V*V)             *
a            DT   B-NP      *       O      -          (A1*           (A1*
special      JJ   I-NP      *       O      -            *               *
edition      NN   I-NP      *       O      -            *A1)            *A1)
around       IN   B-PP      *       O      -       (AM-TMP*             *
noon         NN   B-NP      *       O      -            *AM-TMP)        *
yesterday    NN   B-NP      *       O      -       (AM-TMP*AM-TMP)      *
that         WDT  B-NP    (S*       O      -         (C-A1*         (R-A1*R-A1)
was          VBD  B-VP    (S*       O      -            *               *
filled       VBN  I-VP      *       O      fill         *             (V*V)
entirely     RB   B-ADVP    *       O      -            *         (AM-MNR*AM-MNR)
with         IN   B-PP      *       O      -            *               *
earthquake   NN   B-NP      *       O      -            *            (A2*
news         NN   I-NP      *       O      -            *               *
and          CC   I-NP      *       O      -            *               *
information  NN   I-NP      *S)S)   O      -            *C-A1)          *A2)
.            .    O         *S)     O      -            *               *
```

Figura 5.2: Informação de treino da tarefa conjunta CoNLL 2004. (Fonte: [12])

Foram apresentados dez sistemas com diversas estratégias para abordar a tarefa, entre elas, a utilização de diferentes granularidades de processamento do texto tais

[8] Expressão inglesa que refere um conjunto de palavras.
[9] Do Inglês, *semantic frames*.

como frase a frase [40, 14] ou palavra a palavra [77] e a realização ou não de pós-processamento no final da classificação, de modo a corrigir potenciais erros.

Nas características usadas pelos sistemas também se verificaram diferenças tais como o uso de informação referente a entidades [40, 71], informação das molduras semânticas presentes no *PropBank* [77] e posição relativa do argumento em relação ao verbo [40, 71].

Nos sistemas apresentados verificou-se uma tendência para a utilização de classificadores baseados em máquinas de vectores.

O melhor desempenho foi obtido por Hacioglu *et. al.* [40] com um valor de F_1 igual a 69.49% para o conjunto de teste. O sistema *baseline*, é considerado o sistema básico possuía sete regras específicas da língua Inglesa e o seu valor de F_1 é de 39.87% [12].

O sistema de Hacioglu *et. al.* [40] realiza algum pré-processamento dos dados e as características utilizadas dividem-se em três categorias:

Base. Características que podem ser inferidas directamente a partir do corpus pré-processado: palavras, lemas dos predicados, categorias gramaticais, posição dos sintagmas no método IOB, etiquetas das orações presentes na frase (se está a abrir `(S*` ou a fechar `*S)`) e entidades mencionadas.

Sintagma. Características que podem ser inferidas ao nível dos sintagmas (nível acima das características base): posição em relação ao predicado, caminho entre o sintagma e o predicado expresso nos sintagmas intermédios, dois padrões das orações (indicação das orações entre o sintagma e o predicado e indicação das orações entre o sintagma e o fim ou inicio da frase dependendo da sua posição), indicação se o sintagma está na mesma oração do predicado, sufixos da palavra mais à direita do sintagma, distância do sintagma ao predicado, distância do sintagma ao predicado contando os sintagmas verbais intermédios e tamanho do sintagma expresso em número de palavras.

Frase. Características inferidas ao nível mais geral: categoria gramatical do predicado, frequência do predicado nos dados de treino, contexto do predicado em sintagmas numa janela de tamanho cinco com o predicado na posição central, categoria gramatical da palavra antes e da palavra depois do predicado, molduras semânticas dos predicados presentes no *PropBank* e número de predicados presentes na frase.

Utilizando o método IOB foram treinados 78 classificadores um contra todos[10] (um para cada argumento). Foi utilizada a ferramenta TinySVM[11] [48], uma implementação da máquina de vectores de suporte, com um núcleo polinomial de grau dois.

[10] Do Inglês, *one-vs-all*.
[11] Disponível em `http://chasen.org/~taku/software/TinySVM/`.

Tarefa conjunta do CoNLL 2005. Para esta edição foi utilizado um corpus que inclui o corpus da edição de 2004 aumentado com novas secções do *Penn TreeBank* e o corpus *Brown* [32]. Do corpus *Penn TreeBank* foram utilizadas as secções 2 à 21 para treino, a 24 para desenvolvimento e a 23 para teste. Do corpus *Brown* foram utilizadas três secções (ck01 – ck03) para teste [13].

À semelhança da edição de 2004, a avaliação de resultados foi realizada através das medidas de precisão, cobertura e medida F_1 de cada sistema; os verbos não contaram para a avaliação e um argumento foi considerado correcto quando todas as palavras de um *span* eram identificadas e o argumento classificado correctamente [13].

As classes dos argumentos foram as mesmas propostas na tarefa conjunta do CoNLL 2004 e a informação disponível no corpus também.

Os 19 sistemas participantes abordaram o problema da classificação de várias maneiras: por exemplo, Haghighi *et. al.* [41] utilizaram as estruturas sintácticas fornecidas pelo analisador sintáctico[12] de Charniak [16]; Punyakanok *et. al.* [76] utilizaram o analisador de Charniak em conjunto com o de Collins [20] e os processadores UPC (um analisador gramatical [37] e um identificador de sintagmas e reconhecedor de orações [11]) mas não realizaram pós-processamento; Surdeanu *et. al.* [95] não realizaram pré-processamento com analisadores sintácticos.

As características utilizadas pelos sistemas participantes podem ser divididas em cinco categorias: origem dos dados, argumento, verbo, relação argumento-verbo e preposição [13]:

Origem dos dados. Utilização de analisadores sintácticos e de informação sobre entidades mencionadas. Por exemplo, Punyakanok *et. al.* [76] usou os analisadores de Charniak, Collins e processadores UPC em conjunto com as entidades mencionadas;

Argumento. Tipo de argumentos utilizados; por exemplo, se são usadas todas palavras de um argumento ou só as palavras cabeça[13];

Verbo. Utilização ou não do lema e da categoria gramatical;

Relação argumento-verbo. Considera a distância entre o argumento e o verbo e a posição relativa de um em relação ao outro.

Preposição. Considera a utilização da sequência de argumentos da preposição. Estas características foram usadas apenas pelos sistemas [41, 96].

O método de aprendizagem mais utilizado nesta tarefa conjunta foi o algoritmo de máxima entropia, usado por oito sistemas (entre eles [41, 96]), seguido pelas máquinas de vectores de suporte, usado por seis sistemas (entre eles [73]).

[12] Do Inglês, *parser*.
[13] Do inglês, *head-word* denominada assim por ser a palavra tida principal.

O sistema com melhor pontuação na tarefa conjunta foi o implementado por Punyakanok *et. al.* [76] com valores de F_1 de 79.44%, 67.75% e 77.92% para os conjuntos de teste do *Wall Street Journal*, *Brown* e ambos, respectivamente. Para estes conjuntos, o sistema *baseline* (o mesmo da tarefa conjunta do CoNLL 2004) obteve valores de F_1 de 37.14%, 43.30% e 37.95% [13].

O sistema de Punyakanok *et. al.* [76] utilizou um classificador multi-classe implementado para tarefas de aprendizagem de larga escala. O classificador utiliza a regra de Winnow implementada no SNoW[14] [10] e é composto por quatro fases [76]:

1. Poda da árvore sintáctica de cada frase. Realizada por meio de heurísticas introduzidas em [108] que filtram os argumentos improváveis;

2. Identificação dos argumentos. Realizada por um classificador binário que indica se um candidato é um argumento ou não. Este classificador utiliza um conjunto de 15 características algumas binárias, outras numéricas e outras ainda textuais;

3. Classificação dos argumentos. Realizada por um classificador multi-classe treinado com as etiquetas dos argumentos além da etiqueta `NULL` para reduzir possíveis erros vindos das fases anteriores. Para além das características utilizadas na fase anterior, este classificador utiliza outras 4 características;

4. Inferência. Nesta fase são aplicadas regras linguísticas e estruturais para realizar a correcção do resultado da fase anterior. É considerado um problema de programação linear de inteiros tendo como entrada os valores de confiança de cada tipo de argumento e como resultado a solução óptima que maximiza a soma linear das pontuações de confiança, ou seja, o número esperado de argumentos correctos sujeito a regras definidas para a língua Inglesa.

5.3 Construção do corpus SRL-Publico

Do estudo dos sistemas desenvolvidos no âmbito das tarefas conjuntas de 2004 e 2005 do CoNLL decidiu-se construir, para a língua Portuguesa, um corpus equivalente ao utilizado naquelas tarefas.

Este novo corpus, designado **SRL-Publico**, foi construído a partir da informação extraída dos mesmos documentos utilizados para a construção do corpus POS-Publico (ver secção 4.3). É constituído por 132817 palavras e 4427 frases e utiliza as categorias gramaticais retiradas daquele corpus , informação complementar como lemas, sintagmas, categorias sintácticas e orações, retiradas do Bosque 8.0 e entidades mencionadas obtidas com o sistema de classificação de entidades mencionadas descrito em [62, 63].

[14] Disponível em `http://cogcomp.cs.illinois.edu/page/software_view/1`.

O seu desenvolvimento implicou o seguinte processamento:

1. tratamento dos *hífen* e *underscore* das palavras do Bosque (processo idêntico ao realizado na criação do corpus POS-Publico);

2. os sintagmas e as categorias sintácticas de cada palavra foram extraídas no nível imediatamente abaixo ao início das orações. Foram obtidos os sintagmas mais gerais e as categorias sintácticas referentes ao verbo principal da oração mais exterior. Utilizou-se a representação IOB tanto para os sintagmas como para as categorias sintácticas. A Figura 4.2 mostra que os sintagmas mais exteriores são `np`, `vp`, e `np` e as categorias sintácticas são `SUBJ`, `P`, `SC`;

3. construiu-se a árvore das frases com a indicação do início e fim das orações de cada frase obtidas pelos parêntesis rectos que limitam as orações do tipo `fcl`, `acl` e `icl` (ver Figura 4.2); se existirem orações dentro de orações estes limites também são usados. utilizou-se o método Início-Fim para marcação das orações.

4. obteve-se a informação sobre as entidades através do sistema reconhecedor de entidades mencionadas [62, 63]. Este sistema permite a marcação de três tipos de entidades: Pessoa, Local e Organização.

No final do processamento o corpus, com uma palavra por linha, possui sete colunas: palavra, lema, categoria gramatical, sintagma, categoria sintáctica, entidade mencionada e árvore da oração. A Figura 5.3 mostra um excerto do corpus construído.

```
Vera      Vera      PROP     B-np     B-SUBJ     B-Pessoa     (S*
apagou    apagar    V        B-vp     B-P        O            *
a         o         DET      B-np     B-ACC      O            *
luz       luz       N        I-np     I-ACC      O            *
.         .         PU       O        O          O            *S)
```

Figura 5.3: SRL-Publico: representação da frase '`Vera apagou a luz.`'.

A Tabela 5.1 mostra o número de palavras e *spans* existentes no corpus para cada tipo de sintagma. O significado de cada sigla pode ser consultado na Figura A.5. Na Tabela é possível observar que os sintagmas mais comuns são o sintagma nominal (`np`), seguido do sintagma verbal `vp`; o sintagma menos comum é o adjectival `adjp` com apenas 748 termos.

A Tabela 5.2 mostra o número de palavras e *spans* existentes no corpus para cada tipo de argumento sintáctico. O significado de cada sigla pode ser consultado na Figura A.6. Na Tabela é possível observar que o argumento sintáctico mais comum é o complemento directo (`ACC`) e o menos comum é o predicativo do sujeito/complemento adverbial (`SC/SA`).

Sintagma	*Spans*	Palavras
np	9366	59112
vp	7696	10471
adjp	748	3317
advp	2694	4733
pp	5052	40584

Tabela 5.1: SRL-Publico: nº de palavras e *spans* para cada tipo de sintagma.

Argumento sintáctico	*Spans*	Palavras
P	7268	9373
SUBJ	4673	28718
ACC	3802	37489
SC	1238	11384
PIV	872	8442
ACC-PASS	99	100
SA	351	2064
DAT	94	98
OC	123	889
OA	57	427
SC/SA	1	11

Tabela 5.2: SRL-Publico: nº de palavras e *spans* para cada tipo de argumento sintáctico.

A Tabela 5.3 mostra o número de palavras e *spans* existentes no corpus para cada tipo entidade nomeada.

Entidade nomeada	*Spans*	Palavras
Pessoa	1232	3595
Organização	3412	6401
Local	1459	1551

Tabela 5.3: SRL-Publico: nº de palavras e *spans* para cada tipo de entidade nomeada.

A este corpus poderá ainda ser adicionada informação sobre a análise de dependências analisadas nas edições de 2006 [9] e 2007 [67] das conferências CoNLL, já que na edição de 2006 foi utilizado o Bosque 7.3 para estudo do problema aplicado à língua Portuguesa. Segundo os autores a conversão foi difícil devido à disposição da informação, principalmente a referente aos sintagmas verbais [9].

A utilização deste corpus com informação sobre análise de dependências teria de ser revisto manualmente visto a versão do Bosque usada nesta dissertação diferir da utilizada na edição 2006 do CoNLL. No entanto, esta alteração iria enriquecer o corpus possibilitando o aumento do desempenho do classificador de argumentos sintácticos.

Com essa informação, seriam adicionadas duas novas colunas ao corpus conforme observado na Figura 5.4: (i) identificador da palavra, (ii) palavra, (iii) lema, (iv) categoria gramatical, (v) sintagma, (vi) argumento sintáctico, (vii) entidade men-

cionada, (viii) oração e (ix) dependência.

```
1  Vera     Vera     PROP    B-np    B-SUBJ    B-Pessoa   (S*   2
2  apagou   apagar   V       B-vp    B-P       O          *     0
3  a        o        DET     B-np    B-ACC     O          *     4
4  luz      luz      N       I-np    I-ACC     O          *     2
5  .        .        PU      O       O         O          *S)   2
```

Figura 5.4: SRL-Publico estendido: representação da frase '`Vera apagou a luz`' com informação de dependências.

5.4 O classificador

Para identificar as relações existentes entre entidades foi construído um classificador de argumentos sintácticos preliminar utilizando a ferramenta MinorThird [19] juntamente com o corpus SRL-Publico.

Por outro lado, a utilização do MinorThird com o corpus disponibilizado nas tarefas conjuntas CoNLL de 2004, permite comparar os resultados quer com outra língua (a Inglesa) quer com os classificadores de argumentos sintácticos desenvolvidos no âmbito daquela tarefa.

As sub-secções seguintes introduzem o MinorThird e os corpora utilizados, descrevem a configuração experimental e apresentam e discutem os resultados obtidos.

5.4.1 MinorThird

O MinorThird é uma colecção, em código aberto, de classes implementadas em Java. Foi desenvolvido pelo professor William W. Cohen da Universidade de Carnegie Mellon e actualmente é mantido por Frank Lin [19].

O *toolkit* do MinorThird permite:

- anotar textos tanto manualmente como através de uma aplicação;
- anotar e classificar textos com métodos estado da arte;
- visualizar dados de treino e o desempenho dos classificadores;

O MinorThird usa colecções de documentos para criar uma base de dados denominada *TextBase* e são realizadas afirmações lógicas que são guardadas num objecto do tipo *TextLabels*. Como a anotação presente no objecto *TextLabels* é independente do conteúdo dos documentos, podem existir vários tipos de anotações para o mesmo conjunto de documentos [19].

As anotações enumeram as categorias ou propriedades, podendo ser sintácticas ou semânticas e de uma palavra, documento ou *span*. Estas anotações podem ser criadas

manualmente ou automaticamente através de uma aplicação.

Os métodos de aprendizagem de extracção e classificação de *spans* ou documentos presentes no MinorThird são numerosos. Entre os métodos de aprendizagem sequencial estado-da-arte estão os campos condicionais aleatórios [105] e os modelos condicionais de Markov [93].

5.4.2 Configuração experimental

Para avaliar o desempenho do MinorThird na tarefa de classificação de argumentos sintácticos utilizaram-se dois corpora: um para a língua Portuguesa e outro para a Inglesa. Para a língua Portuguesa foram utilizados os documentos do corpus SRL-Publico (ver secção 4.3; para a língua Inglesa foram utilizados os documentos da tarefa conjunta da edição CoNLL 2004 (ver secção 5.2.2).

Os documentos foram processados para incluir etiquetas XML na marcação dos argumentos sintácticos das frases (coluna 5 do corpus SRL-Publico – Figura 5.3, coluna 7 e posteriores do corpus CoNLL-04 – Figura 5.2).

Para além do Predicado (`P`), apenas foram considerados o Sujeito (`Arg0`) e o Complemento Directo (`Arg1`) por serem os argumentos que aparecem um nº significativo de vezes nos corpora. A Figura 5.5 apresenta uma frase com a marcação proposta.

```
<Arg0>Vera<\Arg0> <P>apagou<\P> <Arg1>a luz<\Arg1>.
```

Figura 5.5: Marcação da frase '`Vera apagou a luz.`' com etiquetas XML.

A Tabela 5.4 apresenta o nº de *spans* marcados em ambos os corpora.

Etiqueta	SRL-Publico	CoNLL-04 (treino)	CoNLL-04 (teste)
`P`	7268	19098	3627
`Arg0`	4673	12709	2579
`Arg1`	3802	18046	3429

Tabela 5.4: Nº de *Spans* dos argumentos sintácticos dos corpora SRL-Publico e CoNLL-04.

Dos diversos testes realizados os dois algoritmos implementados no MinorThird que obtiveram melhores resultados foram o CRF e o SVMCMM. O CRF é uma implementação de campos condicionais aleatórios e o SVMCMM é um implementação de modelos condicionais de Markov treinados com máquinas de vectores de suporte.

Para avaliar o desempenho do classificador utilizou-se o método de validação cruzada 10 pastas para o corpus SRL-Publico e os conjuntos treino-teste para o corpus

CoNLL-04 e calculou-se a precisão (π), cobertura (ρ) e medida F_1 na marcação dos *spans*.

5.4.3 Resultados obtidos

A Tabela 5.5 apresenta os resultados obtidos com os algoritmos SVMCMM e CRF para o corpus SRL-Publico.

	SVMCMM			CRF		
Etiqueta	π	ρ	F_1	π	ρ	F_1
`P`	0.603	0.503	0.548	0.660	0.475	0.545
`Arg0`	0.416	0.283	0.337	0.447	0.237	0.308
`Arg1`	0.285	0.161	0.206	0.361	0.117	0.175

Tabela 5.5: SRL-Publico: Desempenho dos classificadores de argumentos sintácticos.

Observa-se que o algoritmo CRF apresenta melhores valores de precisão, mas o SVMCMM apresenta melhores valores de cobertura. Para ambos os algoritmos os valores da precisão estão pelo menos 0.1 acima dos da cobertura (para o algoritmo CRF o `Arg0` e o `Arg1` apresentam valores de precisão superiores em 0.2 quando comparados com os da cobertura).

A etiqueta que obteve o melhor desempenho foi o Predicado (`P`), com valores de F_1 acima dos 54%. O Complemento Directo (`Arg1`) obteve o pior desempenho com valores de F_1 pouco superiores a 20% para o algoritmo SVMCMM e abaixo dos 18% para o CRF. O Sujeito (`Arg0`) obteve um desempenho com valores de F_1 acima dos 33% para o algoritmo SVMCMM e inferior a 31% para o CRF.

Estes valores permitem inferir que um classificador sequencial usando apenas informação sintáctica quando comparado com o sistema de [6] que utiliza gramáticas possui valores de F_1 inferiores em 30%. Este facto permite concluir que é necessária mais informação linguística e uma nova abordagem para obter um desempenho estado da arte.

A Tabela 5.6 apresenta o desempenho dos algoritmos SVMCMM e CRF para o corpus CoNLL-04.

	SVMCMM			CRF		
Etiqueta	π	ρ	F_1	π	ρ	F_1
`P`	0.850	0.823	0.836	0.842	0.805	0.823
`Arg0`	0.599	0.464	0.523	0.699	0.463	0.557
`Arg1`	0.372	0.170	0.234	0.414	0.151	0.221

Tabela 5.6: CoNLL-04: Desempenho dos classificadores de argumentos sintácticos.

Para o corpus CoNLL-04, o desempenho obtido por ambos os algoritmos é equivalente. Mais uma vez o Predicado (`P`) apresenta os melhores resultados com valores de F_1 acima dos 82%; em contra-partida, o Complemento Directo (`Arg1`) apresenta valores de F_1 próximos dos 23%.

Comparando as Tabelas 5.5 e 5.6 é possível concluir que o desempenho obtido com o corpus SRL-Publico é muito inferior ao obtido com o corpus CoNLL-04. Uma possível explicação para esta diferença poderá ser o tamanho dos corpora: o treino com o corpus CoNLL-04 possui sensivelmente o triplo dos exemplos relativamente ao corpus SRL-Publico; outra explicação é a estrutura sintáctica da língua Inglesa ser mais simples que a da língua Portuguesa.

A Tabela 5.7 apresenta os valores de F_1 obtidos com o MinorThird (SVMCMM) e com o 'melhor' ([40]) e 'pior' ([106]) sistemas da tarefa conjunta CONLL 2004 [12]. Apresentam-se apenas os resultados das etiquetas `Arg0` e `Arg1`, uma vez que o Predicado não foi avaliado na tarefa conjunta.

Etiqueta	MinorThird	[40]	[106]
`Arg0`	0.523	0.814	0.562
`Arg1`	0.234	0.716	0.490

Tabela 5.7: Valores de F_1 obtidos pelo MinorThird e os sistemas [40] e [106] da tarefa conjunta CoNLL 2004.

Dos resultados é possível concluir que o uso de informação linguística adicional como por exemplo categorias gramaticais, sintagmas, orações e entidades mencionadas é útil para a tarefa de classificação de argumentos sintácticos.

Capítulo 6

Conclusões e trabalho futuro

No âmbito desta dissertação desenvolveram-se recursos e analisou-se o desempenho de algumas ferramentas para processamento da língua Portuguesa:

- com base no Bosque 8.0 e informação obtida de outros recursos e ferramentas criaram-se dois corpora:
  - o POS-Publico para etiquetar categorias gramaticais;
  - o SRL-Publico para classificar argumentos sintácticos;
- aplicando técnicas de aprendizagem supervisionada (sistemas orientados a dados) desenvolveram-se dois modelos:
  - um etiquetador de categorias gramaticais com a ferramenta SVMTool;
  - um classificador preliminar de argumentos sintácticos com a ferramenta MinorThird;
- a utilização do classificador de argumentos sintácticos desenvolvido em conjunto com um reconhecedor de entidades possibilitou a **extracção de relações entre entidades** presentes num texto.

A próxima secção apresenta as principais conclusões e a seguinte ideias para trabalho futuro.

6.1 Conclusões

Neste trabalho, direccionado para a língua Portuguesa, estudaram-se alguns problemas do processamento da língua natural: a atribuição de categorias gramaticais às

palavras e a classificação dos argumentos sintácticos das frases.

As conclusões podem ser separadas nos temas abordados durante a realização do trabalho:

Aprendizagem de Língua Natural. Como este trabalho pretendia utilizar sistemas orientados a dados estudaram-se os algoritmos de aprendizagem supervisionada estado-da-arte para os problemas em estudo, e as técnicas para marcação de corpora e avaliação de resultados. Por outro lado, analisaram-se as tarefas que têm sido abordadas no âmbito das conferências CoNLL que estudam tópicos de aprendizagem de língua natural.

Extracção de relações entre entidades. Para identificar as relações existentes entre entidades, criou-se um sistema que combina a informação obtida pela extracção das entidades com aquela obtida pela classificação dos argumentos sintácticos das frases; o resultado pode ser observado na Figura 3.4.

Etiquetador de categorias gramaticais. Para obter a categoria gramatical das palavras necessária, quer para a extracção de entidades quer para a classificação de argumentos sintácticos, foi desenvolvido um etiquetador de categorias gramaticais.

Foram estudadas várias ferramentas em código aberto orientadas a dados, nomeadamente o FreeLing e o SVMTool, tendo sido escolhido o último principalmente devido à sua precisão, eficiência e robustez.

Para criar o modelo do etiquetador foi desenvolvido um corpus para utilização com o SVMTool – o POS-Publico. Este corpus foi criado a partir de um sub-conjunto do *CETEMPúblico* do corpus Bosque 8.0 e utiliza a classificação gramatical do léxico *LABEL-LEX*. As características deste corpus estão presentes na Tabela 4.4.

O etiquetador obteve valores médios de 96% para a precisão, 94% para a cobertura e 95% para a medida F_1 (Tabela 4.6). Verificou-se que os valores obtidos são equivalentes aos reportados para o etiquetador `LX-Suite` e que o valor de F_1 é muito próximo da barreira dos 97% falada em [24].

Com base nestes valores é possível concluir que o etiquetador de categorias gramaticais desenvolvido possui um desempenho equivalente a outros etiquetadores comerciais.

Classificador de argumentos sintácticos. Com o estudo das tarefas conjuntas das conferências CoNNL 2004 e 2005, decidiu-se construir um corpus semelhante ao utilizado naquelas tarefas para a língua Portuguesa – o corpus SRL-Publico. Este corpus utiliza os mesmos documentos do corpus POS-Publico (retirados da secção *CETEM-Publico* do Bosque 8.0) e inclui a seguinte informação linguística: palavras, lemas, sintagmas, orações, categorias gramati-

cais, entidades mencionadas e argumentos sintácticos. As características deste corpus estão presentes nas Tabelas 5.1 e 5.2.

À semelhança do etiquetador de categorias gramaticais, o classificador de argumentos sintácticos foi desenvolvido com recurso a uma ferramenta orientada a dados – o MinorThird, tendo sido obtidos modelos tanto para a língua Portuguesa como para a Inglesa.

O MinorThird é um gerador de classificadores sequenciais desenvolvido em código aberto para as tarefas de classificação de textos e extracção de entidades. Por esta razão, o modelo desenvolvido para a língua Inglesa (com o corpus CoNLL-04) obteve valores de desempenho abaixo dos obtidos por sistemas desenvolvidos especificamente para esta tarefa como os apresentados nas tarefas conjuntas das conferências CoNLL de 2004 (ver Tabela 5.7).

O desempenho do classificador para a língua Portuguesa (ver Tabela 5.5), construído com base no corpus SRL-Publico, é inferior àquele obtido para a língua Inglesa. As hipóteses sugeridas para esta diferença residem no tamanho dos corpora (o corpus CoNLL é cerca de três vezes maior que o SRL-Publico) e/ou à existência de estruturas sintácticas mais complexas e um número de contracções superiores na língua Portuguesa quando comparada com a Inglesa.

Ao comparar o desempenho do classificador para a língua Portuguesa obtido com o MinorThird com o classificador documentado em [6] verifica-se que os valores são muito díspares. Isto é mais uma indicação da necessidade de implementação dum classificador com mais informação linguística utilizando, por exemplo, o corpus SRL-Publico.

O classificador de Bick [6] obteve um valor de F_1 igual a 88.6%. Quando comparado com as edições de 2008 [94] e 2009 [42] das conferências CoNLL, verificou-se que o seu desempenho é superior. Na edição de 2008 o melhor desempenho foi o obtido com o classificador de Nugues *et. al* [45] com um valor de F_1 igual a 85.95% (utiliza um classificador que realiza uma classificação conjunta de dependências e de argumentos sintácticos); quando utilizado apenas na tarefa de classificação de argumentos obteve um valor de 81.75%. Já na edição de 2009, em que foram usadas 7 línguas diferentes, a melhor média dos desempenhos foi obtida pelo sistema de Che *et. al* [17], com um valor de F_1 igual a 82.64% para a tarefa de classificação de dependências e argumentos; para a classificação de argumentos obteve um valor médio de F_1 de 79.94%.

6.2 Trabalho futuro

Existem várias vertentes a seguir como trabalho futuro. Uma das mais importantes é desenvolver um classificador de argumentos sintácticos orientado a dados que, à semelhança dos desenvolvidos nas tarefas conjuntas das conferências CoNLL faça uso de informação linguística quer das palavras como dos sintagmas e orações, nomeada-

mente através da utilização do corpus desenvolvido no âmbito desta dissertação – o SRL-Publico.

O desenvolvimento de tal classificador permitirá fazer uma análise mais justa quer a nível do corpus SRL-Publico comparando os resultados obtidos pelo classificador com outros corpora, quer a nível do desempenho do próprio classificador comparando os seus resultados com os obtidos por outros sistemas (e mesmo corpus).

Por outro lado, é possível melhorar o corpus SRL-Publico. Este desenvolvimento poderá ser feito através da adição de mais documentos (como por exemplo a componente *CETEM-Folha* do Bosque 8.0) e/ou acrescentando informação referente às dependências funcionais das palavras. Esta informação pode ser obtida a partir do corpus utilizado na tarefa conjunta do CoNLL 2006 em que a língua Portuguesa foi abrangida (no entanto, esta informação terá de ser revista manualmente já que o corpus usado no CoNLL 2006 é uma versão anterior à utilizada neste trabalho).

A criação de um classificador de argumentos sintácticos com melhor desempenho irá melhorar o sistema que extrai as relações entre entidades, identificando novas relações entre entidades presentes no texto, já que os valores de cobertura do sistema preliminar desenvolvido são muito baixos.

Por outro lado, a utilização de outros sistemas de reconhecimento de entidades que não sofram dos problemas do sistema reconhecedor utilizado, documentado em [62, 63], irá também beneficiar o sistema de extracção de relações entre entidades. Um possível sistema a testar poderá ser o MinorThird.

Bibliografia

[1] Aberdeen, J., Burger, J., Day, D., Hirschman, L., Robinson, P., and Vilain, M. Mitre: description of the alembic system used for muc-6. In *MUC6 '95: Proceedings of the 6th conference on Message understanding* (Morristown, NJ, USA, 1995), Association for Computational Linguistics, pp. 141–155.

[2] Abreu, C. Internet em números em 2010. http://aeiou.expresso.pt/internet-em-numeros-em-2010=f622333, Dezembro 2010.

[3] Afonso, S., Bick, E., Haber, R., and Santos, D. Florestra sintá(c)tica: A treebank for portuguese. In *LREC'02 – 3rd International Conference on Language Resources and Evaluation* (Las Palmas, Spain, May 2002), M. G. Rodrigues and C. P. S. Araujo, Eds., ELRA, pp. 1698–1703.

[4] Barreto, F., Branco, A., Ferreira, E., Mendes, A., Bacelar do Nascimento, M. F. P., Nunes, F., and Silva, J. Open resources and tools for the shallow processing of portuguese. In *LREC'06 – 6th International Conference on Language Resources and Evaluation* (2006).

[5] Bick, E. *The Parsing System "PALAVRAS": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Aarhus University Press, 2000.

[6] Bick, E. Automatic semantic-role annotation for portuguese. In *SBC'07 – XXVII Congresso da Sociedade Brasileira de Computação* (2007).

[7] Branco, A., and Silva, J. Evaluating solutions for the rapid development of state-of-the-art pos taggers for portuguese. In *LREC'04 – 4th International Conference on Language Resources and Evaluation* (2004), pp. 507–510.

[8] Brants, T. Tnt - a statistical part-of-speech tagger. In *ANLP'00 – 6th Applied Natural Language Processing* (2000).

[9] Buchholz, S., and Marsi, E. Conll-x shared task on multilingual dependency parsing. In *CoNLLX'06 – 10th Conference on Computational Natural Language Learning* (2006), pp. 149–164.

[10] Carlson, A., Cumby, C., Rizzolo, N., Roth, D., and Rosen, J. Winnow. http://cogcomp.cs.illinois.edu/software/doc/snow-userguide/node9.html, 2004.

[11] Carreras, X., and Màrquez, L. Phrase recognition by filtering and ranking with perceptrons. In *RANLP'03 – Recent Advances in Natural Language Processing* (2003), pp. 205–216.

[12] Carreras, X., and Màrquez, L. Introduction to the conll-2004 shared task: Semantic role labeling. In *CoNLL'04 – 8th Conference on Computational Natural Language Learning* (2004).

[13] Carreras, X., and Màrquez, L. Introduction to the conll-2005 shared task: Semantic role labeling. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (2005).

[14] Carreras, X., Màrquez, L., and Chrupala, G. Hierarchical recognition of propositional arguments with perceptrons. In *CoNLL'04 Shared Task – 8th Conference on Computational Natural Language Learning* (2004).

[15] Chang, C., and Lin, C. *LIBSVM: a library for support vector machines*, 2001.

[16] Charniak, E. A maximum-entropy inspired parser. In *NAACL'00 – North American Chapter of the Association for Computational Linguistics* (2000).

[17] Che, W., Li, Z., Li, Y., Guo, Y., Qin, B., and Liu, T. Multilingual dependency-based syntactic and semantic parsing. In *CoNNL'09 Shared Task – 13th Conference on Computational Natural Language Learning* (2009), Association for Computational Linguistics, pp. 49–54.

[18] Chinchor, N., Robinson, P., and Brown, E. Hub-4 named entity task definition. In *DARPA Broadcast News Workshop* (1998).

[19] Cohen, W. Minorthird: methods for identifying names and ontological relations in text using heuristics for inducing regularities from data, 2004.

[20] Collins, M. Head-driven statistical models. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.5.5668&rank=1, 1999.

[21] Crouch, D., Dalrymple, M., Kaplan, R., King, T., Maxwell, J., and Newman, P. Xle documentation. http://www2.parc.com/isl/groups/nltt/xle/, 2008.

[22] Darroch, J., and Ratcliff, D. Generlized iterative scaling for log-linear models. *The Annals of Mathematical Statistics 43*, 5 (1972), 1470–1480.

[23] De Meulder, F., and Sang, E. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *CoNLL'03 – 7th Conference on Computational Natural Language Learning* (2003), W. Osborne, Ed., pp. 142–147.

[24] Dickinson, M. An investigation into improving part-of-speech tagging. In *Proceedings of the Fifth Workshop on Treebanks and Linguistic Theories (TLT 2006)* (2006).

[25] Dietterich, T. Machine learning. *Nature Encyclopedia of Cognitive Science* (2003).

[26] Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., and Weischedel, R. The automatic content extraction (ace) program–tasks, data, and evaluation. In *Proceedings of LREC 2004* (2004), pp. 837–840.

[27] Dow Jones & Company. The wall street journal. http://europe.wsj.com/home-page, 1889.

[28] Duran, M. S., Ramisch, C., Aluísio, S. M., and Villavicencio, A. Identifying and analyzing brazilian portuguese complex predicates. In *MWE'11 – Workshop on Multiword Expressions: from Parsing and Generation to the Real World* (Junho 2011).

[29] Edisport S.A. Cofina Media Grupo Cofina. Record. http://www.record.xl.pt, 1948.

[30] Empresa Folha da Manhã S.A. Folha.com. http://www.folha.uol.com.br, 1921.

[31] Farkas, R., Vincze, V., Móra, G., Csirik, J., and Szarvas, G. The conll-2010 shared task: learning to detect hedges and their scope in natural language text. In *CoNLL '10: Shared Task Proceedings of the Fourteenth Conference on Computational Natural Language Learning* (2010), Association for Computational Linguistics.

[32] Francis, W., and Kucera, H. Manual of information to accompany a standard corpus of present-day edited american english, for use with digital computers. http://icame.uib.no/brown/bcm.html, 1997.

[33] Freitas, C., and Afonso, S. Bíblia florestal: Um manual lingüístico da floresta sintá(c)tica. http://www.linguateca.pt/Floresta/BibliaFlorestal, Setembro 2008.

[34] Ghahramani, Z. *Unsupervised Learning*, vol. 3176/2004. Springer, 2004.

[35] Gildea, D., and Hockenmaier, J. Identifying semantic roles using combinatory categorial grammar. In *EMNLP '03 Proceedings of the 2003 conference on Empirical methods in natural language processing* (2003), Association for Computational Linguistics, pp. 57–64.

[36] Gildea, D., and Jurafsky, D. Automatic labeling of semantic roles. *Computational Linguistics 28* (2002), 245–288.

[37] Giménez, J., and Màrquez, L. Fast and accurate part-of-speech tagging: The svm aproach revisited. In *RANLP'03 – Recent Advances in Natural Language Processing* (2003).

[38] Giménez, J., and Márquez, L. Svmtool: A general pos tagger generator based on support vector machines. In *LREC'04 – 4th International Conference on Language Resources and Evaluation* (2004).

[39] Grishman, R., and Sundheim, B. Message understanding conference-6: a brief history. In *Proceedings of the 16th conference on Computational linguistics - Volume 1, COLING'96* (1996), Association for Computational Linguistics, pp. 466–471.

[40] Hacioglu, K., Pradhan, S., Ward, W., Martin, J., and Jurafsky, D. Semantic role labeling by tagging syntactic chunks. In *CoNLL'04 Shared Task – 8th Conference on Computational Natural Language Learning* (2004), pp. 110–113.

[41] Haghighi, A., Toutanova, K., and Manning, C. A joint model for semantic role labeling. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (Junho 2005).

[42] Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Marquez, L., Meyers, M., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., and Zhang, Y. The conll 2009 shared task: Syntactic and semantic dependencies in multiple languages. In *CoNNL'09 Shared Task – 13th Conference on Computational Natural Language Learning* (2009), pp. 1–18.

[43] Hendrickx, I., Mendes, A., Pereira, S., Goncalves, A., and Duarte, I. Complex predicates annotation in a corpus of portuguese. In *ACL'10 – 4th ACL Linguistic Annotation Workshop* (2010), pp. 100–108.

[44] Joachims, T. Making large-scale svm learning practical. *Advances in KernelMethods - Support Vector Learning* (1999).

[45] Johansson, R., and Nugues, P. Dependency based syntactic–semantic analysis with propbank and nombank. In *CoNNL'08 – 12th Conference on Computational Natural Language Learning* (2008).

[46] Kingsbury, P., and Palmer, M. From treebank to propbank. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.7566, 2002.

[47] Kubler, S., McDonald, R., and Nivre, J. *Dependency Parsing*. Morgan & Claypool, Janeiro 2009.

[48] Kudo, T. Tinysvm: Support vector machines. http://chasen.org/~taku/software/TinySVM, 2002.

[49] Kudoh, T., and Matsumoto, Y. Use of support vector learning for chunk identification. In *CoNLL'00 – 4th Conference on Computational Natural Language Learning* (2000), pp. 142–144.

[50] Laboratório de Engenharia da Linguagem, IST. Label-lex. http://label.ist.utl.pt/pt/apresentacao.php, 1995.

[51] Lafferty, J., McCallum, A., and Pereira, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML'01 – 18th International Conference on Machine Learning* (2001), pp. 282–289.

[52] Lin, D. Minipar home page. http://webdocs.cs.ualberta.ca/~lindek/minipar.htm.

[53] Linguateca. Ferramentas computacionais para o português. http://www.linguateca.pt/ferramentas_info.html#fer2pln, 2009.

[54] Linguateca. Florestra sintá(c)tica. http://www.linguateca.pt/floresta/corpus.html, 2009.

[55] Liu, T., Che, W., Li, S., Hu, Y., and Liu, H. Semantic role labeling system using maximum entropy classifier. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (2005), pp. 189–192.

[56] Lorena, A., and Carvalho, A. Introdução às máquinas de vectores suporte (support vector machines). *Relatórios técnicos do ICMC* (Abril 2003).

[57] Lorena, A., and Carvalho, A. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada - RITA XIV*, 2 (2007), 43–67.

[58] Marcus, M., Santorini, B., and Marcinkiewicz, M. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics 19*, 2 (1993), 313–330.

[59] Marcus, M., Taylor, A., MacIntyre, R., Bies, A., Cooper, C., Ferguson, M., and Littman, A. The penn treebank project. http://www.cis.upenn.edu/~treebank/, 1999.

[60] McCallum, A., Freitag, D., and Pereira, F. Maximum entropy markov models for information extraction and segmentation. In *Proceeding 17th International Conference on Machine Learning* (2000), Morgan Kaufmann, pp. 591–598.

[61] Microsoft. Microsoft. http://www.microsoft.com/pt/pt/default.aspx, 1975.

[62] Miranda, N., Raminhos, R., Seabra, P., Sequeira, J., Gonçalves, T., and Quaresma, P. Reconhecimento de entidades nomeadas com svm. In *Actas das Jornadas de Informática da Universidade de Évora 2010* (Novembro 2010).

[63] Miranda, N., Raminhos, R., Seabra, P., Sequeira, J., Gonçalves, T., and Quaresma, P. Named entity recognition using machine learning techniques. In *EPIA'11 – 15th Portuguese Conference on Artificial Intelligence* (Lisbon, PT, October 2011 *to be published*).

[64] Mitchell, T. M. *Machine Learning*. McGraw-Hill, 1997.

[65] Moschitti, A. Efficient convolution kernels for dependency and constituent syntactic trees. In *Proc. 17th Eur. Conf. Mach. Learn* (2006).

[66] Nadeau, D., and Sekine, S. A survey of named entity recognition and classification. *Linguisticae Investigationes 30*, 1 (2007), 3–26.

[67] Nivre, J., Hall, J., Kubler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. The conll 2007 shared task on dependency parsing. In *CoNNL'07 Shared Task – 11th Conference on Computational Natural Language Learning* (Junho 2007), Association for Computational Linguistics, pp. 915–932.

[68] Osborne, M., and Sang, E. Noun phrase detection by repeated chunking. In *Workshop on Computational Natural Language Learning (CoNLL'99)* (1999), Association for Computational Linguistics.

[69] Padró, L., Collado, M., Reese, S., Lloberes, M., and Castellón, I. Freeling 2.1: Five years of open-source language processing tools. In *LREC'10 – 7th Language Resources and Evaluation Conference* (2010).

[70] Palmer, M., Gildea, D., and Kingsbury, P. The preposition bank: An annotated corpus of semantic roles. *Computational Linguistics 31* (2005).

[71] Park, M., Park, K., Hwang, Y., and Rim, H. Two-phase semantic role labelling based on support vectos machines. In *CoNLL'04 Shared Task – 8th Conference on Computational Natural Language Learning* (2004).

[72] Pedras, J., and Quaresma, P. Extracção de informação de documentos em língua portuguesa: aplicação a domínios de anúncios. In *Actas das Jornadas de Informática da Universidade de Évora 2010* (2010).

[73] Pradhan, S., Hacioglu, K., Ward, W., Martin, J., and Jurafsky, D. Semantic role chunking combining complementary syntactic views. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (2005).

[74] Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Procceedings of CoNLL 2011* (2011), Association for Computational Linguistics.

[75] PÚBLICO Comunicação Social S.A. Público. http://www.publico.pt, 1990.

[76] Punyakanok, V., Koomen, P., Roth, D., and Yih, W. Generalized inference with multiple semantic role labeling systems. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (2005), pp. 181–184.

[77] Punyakanok, V., Roth, D., Yih, W., Zimak, D., and Tu, Y. Semantic role labeling via generalized inference over classifiers. In *CoNLL'04 Shared Task – 8th Conference on Computational Natural Language Learning* (2004).

[78] QREN. Qren - quadro de referência estratégico nacional. http://www.qren.pt/.

[79] Rabiner, L. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE 77*, 2 (1989), 257–286.

[80] Rabiner, L., and Juang, B. An introduction to hidden markov models. *IEEE ASSP Magazine* (Janeiro 1986).

[81] Ramshaw, L., and Marcus, M. Text chunking using transformation-based learning. In *3rd Workshop on Very Large Corpora* (1995), pp. 82–94.

[82] Ratnaparkhi, A. A maximum entropy model part-of-speech tagger. In *EMNLP'96 – Empirical Methods in Natural Language Processing Conference* (1996), pp. 133–141.

[83] Roy, G. Semantic role labeling. Tech. rep., Indian Institute of Technology Bombay, Abril 2010.

[84] Sang, E. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *CoNLL'02 – 6th Conference on Computational Natural Language Learning* (2002), pp. 155–158.

[85] Sang, E., and Buchholz, S. Introduction to the conll-2000 shared task: Chunking. In *CoNLL'00 – 4th Conference on Computational Natural Language Learning* (2000), pp. 127–132.

[86] Sang, E., and Déjean, H. Introduction to the conll-2001 shared task: Clause identification. In *CoNLL'01 – 5th Conference on Computational Natural Language Learning* (2001), pp. 53–57.

[87] Schmid, H. Probabilistic part-of-speech tagging using decision trees. In *ICNMLP'94 – International Conference on New Methods in Language Processing* (1994).

[88] Schmid, H. Improvements in part-of-speech tagging with an application to german. In *EACL'95 – SIGDAT Workshop: From Text to Tags* (1995).

[89] Sekine, S., and Eriguchi, Y. Japanese named entity extraction evaluation: analysis of results. In *Proceedings of the 18th conference on Computational linguistics - Volume 2, COLING'00* (2000), Association for Computational Linguistics, pp. 1106–1110.

[90] Sequeira, J., Gonçalves, T., and Quaresma, P. Semantic role labeling for portuguese – a preliminary approach. In *PROPOR'12 – International Conference on Computational Processing of the Portuguese Language* (2012), (*submitted*).

[91] SIGNLL. Conll: the conference of signll. http://ifarm.nl/signll/conll, Dezembro 2010.

[92] Sociedade Editorial S.A. Megafin. Oje. http://www.oje.pt, 2008.

[93] Stamp, M. A revealing introduction to hidden markov models. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.136.137&rank=1, 2004.

[94] Surdeanu, M., Johansson, R., Meyers, M., Marquez, L., and Nivre, J. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *CoNNL'08 – 12th Conference on Computational Natural Language Learning* (2008), pp. 159–177.

[95] Surdeanu, M., and Turmo, J. Semantic role labeling using complete syntactic analysis. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (2005), pp. 221–224.

[96] Sutton, C., and Mccallum, A. Joint parsing and semantic role labeling. In *CoNLL'05 – 9th Conference on Computational Natural Language Learning* (2005), pp. 225–228.

[97] Tan, P., Steinbach, M., and Kumar, V. *Introduction to Data Mining-Tan2005*. Addison Wesley, 2005.

[98] Temperley, D., Sleator, D., and Lafferty, J. Link grammar. http://www.link.cs.cmu.edu/link/, 2009.

[99] Toutanova, K., Klein, D., Manning, C., and Singer, Y. Feature-rich part-of-speech tagging using a cyclic dependency network. In *HLT-NAACL'03 – Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics* (2003), pp. 252–259.

[100] Universidade de Évora. Universidade de évora. http://www.uevora.pt.

[101] Vapnik, V. *Statistical Learning Theory*. Wiley-Interscience, Setembro 1998.

[102] Viatecla. Viatecla. http://www.viatecla.pt.

[103] Voutilainen, A. A syntax-based part-of-speech analyser. In *EACL'95 – 7th Conference of the European Chapter of the Association for Computational Linguistics* (1995), pp. 157–164.

[104] Wallach, H. Efficient training of conditional random fields. Master's thesis, School of Cognitive Science, 2002.

[105] Wallach, H. Conditional random fields: An introduction. http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.124.6711, 2004.

[106] Williams, K., Dozier, C., and McCulloh, A. Learning transformation rules for semantic role labeling. In *CoNLL'04 Shared Task – 8th Conference on Computational Natural Language Learning* (2004), pp. 134–137.

[107] Worldometers. Worldometers. http://www.worldometers.info, 2011.

[108] Xue, N., and Palmer, M. Calibrating features for semantic role labeling. In *Proc. of the EMNLP-2004* (2004), pp. 88–94.

[109] Zhang, M., Che, W., GuoDong, Z., Aw, A., Tan, C., Liu, T., and Li, S. Semantic role labeling using a grammar-driven convolution tree kernel. *IEEE Transactions on Audio, Speech, and Language Processing 16*, 7 (September 2008), 1315–1329.

Anexos

Anexo A

Glossário e números do Bosque

As tabelas que compõem este anexo fornecem a informação sobre as diversas etiquetas que compõem o corpus Bosque 8.0 e algumas contagens que o caracterizam. Inclui-se os argumentos ao nível das orações (Figura A.1), as formas de grupo (Figura A.2), os tipos de orações (Figura A.3), as categorias gramaticais (Figura A.4), e as contagens de orações (Figura A.5), sintagmas (Figura A.6), classes de palavras (Figura A.7) e funções das palavras nas orações (Figura A.8).

símbolo	categoria
SUBJ	sujeito (incluindo sujeitos impessoais *se*)
ACC	objecto directo (incluindo alguns tipos de *se*)
ACC-PASS	função do clítico *se* numa oração passiva (partícula apassivante)
DAT	objecto indirecto pronominal (incluindo *se*)
PIV	objecto preposicional
SA	complemento adverbial [pode ser substituído por um pronome adverbial] (relativo ao sujeito)
OA	complemento adverbial (relativo ao objecto)
SC	predicativo do sujeito
OC	predicativo do objeto
COM	complementizador em estruturas de comparação (*como*, (*do*) *que*)
P	predicador
PMV	em contexto de coordenação, verbo principal coordenado com os seus próprios constituintes
PAUX	em contexto de coordenação, verbo auxiliar partilhado por verbos principais com os seus próprios constituintes
MV	verbo principal
AUX	verbo auxiliar
AUX<	em contexto de coordenção, partícula de ligação entre o auxiliar partilhado e verbos coordenados
PRT-AUX	partícula de ligação verbal

Figura A.1: Notação e descrição dos argumentos ao nível das orações. (Fonte: [54])

símbolo	categoria
np	sintagma nominal (H: nome or pronome)
adjp	sintagma adjectival (H: adjectivo ou determinante)
advp	sintagma adverbial (H: advérbio)
vp	sintagma verbal (contém sempre MV e poderá exibir AUX)
pp	sintagma preposicional (H: preposição)
cu	sintagma evidenciador de relação de coordenação
sq	sequência de funções discursivas; sequência de elementos identificadores do falante, tema, etc. e do discurso propriamente dito

Figura A.2: Notação e descrição das formas de grupo (sintagmas). (Fonte: [54])

símbolo	categoria
fcl	oração finita
icl	oração não-finita
acl	oração averbal

Figura A.3: Notação e descrição dos tipos de orações. (Fonte: [54])

símbolo		categoria
n		nome, substantivo
prop		nome próprio
adj		adjectivo
n-adj		flutuação entre substantivo e adjectivo
v	v-fin	verbo finito
	v-inf	infinitivo
	v-pcp	particípio
	v-ger	gerúndio
art		artigo
pron	pron-pers	pronome pessoal
	pron-det	pronome determinativo
	pron-indp	pronome independente (com comportamento semelhante ao nome)
adv		advérbio
num		numeral
prp		preposição
intj		interjeição
conj	conj-s	conjunção subordinativa
	conj-c	conjunção coordenativa

Figura A.4: Notação e descrição das categorias gramaticais. (Fonte: [54])

Formas oracionais	etiqueta	quantidade
finitas	fcl	15.573
não finitas	icl	5.704
averbais	acl	774
total		**22.051**

Figura A.5: Contagem das orações. (Fonte: [54])

Sintagmas	etiqueta	quantidade
nominais	np	61.537
adjectivais	adjp	9.608
adverbiais	advp	6.505
verbais	vp	21.747
preposicionais	pp	32.949
evidenciador coordenação	cu	5.511
sequências discursivas	sq	125
total		**137.982**

Figura A.6: Contagem dos sintagmas. (Fonte: [54])

Classes de palavras	etiqueta	quantidade
substantivos	n	40.728
substantivos/adjectivos	n-adj	633
adjectivos	adj	10.424
nomes próprios	prop	11.977
advérbios	adv	9.251
verbos	v-.*	26.537
finitos	v-fin	*15.877*
gerúndios	v-ger	*863*
particípios	v-pcp	*4.730*
infinitivos	v-inf	*5.067*
artigos	art	29.793
pronomes	pron-.*	11.147
determinativos	pron-det	*5.054*
independentes	pron-rel	*3.309*
pessoais	pron-pess	*2.784*
advérbios	adv	9.251
preposições	prp	33.048
interjeições	intj	39
conjunções	conj-.*	7.506
subordinativa	conj-s	*2.328*
coordenativa	conj-c	*5.178*
prefixos	ec	179
total		**190.513**

Figura A.7: Contagem das classes de palavras. (Fonte: [54])

função	etiqueta	quantidade
sujeito	SUBJ	29.793
obj. directo	ACC	11.279
part. apassivante	ACC-PASS	206
obj. ind. pronominal	DAT	236
obj. ind. preposicional	PIV	2.773
agente passiva	PASS	744
adj. adverbiais	...	17.435
do sujeito	SA	*848*
do objecto	OA	*173*
livres	ADVL	*16.414*
predicativos	...	4.027
do sujeito	SC	*3.364*
do objecto	OC	*367*
verbo-nominais	PRED	*296*
vocativo	VOC	29
apostos	...	4.815
normal	APP	*802*
epit. predicativo	N<PRED	*3.898*
da oração	>S, S<	*115*
compl. nominais	N<ARG.*	2.048
do sujeito	N<ARG1	*258*
do objecto	N<ARG2	*1.292*
outros	N<ARG	*498*
predicador	P	21.385
foco	FOC	221
tópico	TOP	8

Figura A.8: Contagem das funções das palavras existentes nas orações. (Fonte: [54])

Anexo B

Categorias gramaticais do Label-Lex

Esta tabela enumera as principais categorias gramaticais presentes no Label-Lex.

Grammatical categories (POS)		
Notation	*POS*	*Example*
ADJ	Adjective	abatida,abatido.**ADJ**+Pd+z1:fs
ADV	Adverb	agora,agora.**ADV**+z1
CONJ	Conjunction	ou,ou.**CONJ**+z1
DET	Determiner	a,o.**DET**+Art+Def+z1:fs
INTERJ	Interjection	ui,ui.**INTERJ**+z1
N	Noun	abade,abade.**N**:ms
PREP	Preposition	de,de.**PREP**
PRO	Pronoun	ela,eu.**PRO**+Pes+N+z1:3fs
V	Verb	abro,abrir.**V**+z1:P1s
PFX	Prefix	hiper,hiper.**PFX**+z1
Contractions		
PREPXDET	Preposition_Determiner	deste,deste.**PREPXDET**+Dem+z1:ms
PREPXPRO	Preposition_Pronoun	à,ao.**PREPXPRO**+Dem+z1:fs
PREPXADV	Preposition_Adverb	daqui,daqui.**PREPXADV**+z1

Figura B.1: Notação das principais categorias gramaticais presentes no Label-Lex. (Fonte: [50])